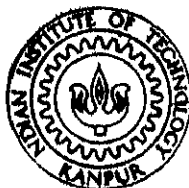


FINITE DECONVOLUTION DATA RECOVERY SCHEMES FOR BLOCK DATA COMMUNICATION OVER DISPERSIVE CHANNELS

by

POONACHA P G

E E
1986
P
P00
FIN



DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY KANPUR
OCTOBER 1986

FINITE DECONVOLUTION DATA RECOVERY SCHEMES FOR
BLOCK DATA COMMUNICATION OVER DISPERSIVE CHANNELS

A Thesis Submitted
In Partial Fulfilment of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

by
POONACHA P G

to the
DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY KANPUR
OCTOBER 1986

17 DEC 1987
CENTRAL LIBRARY
I 7 K

A N A 99165
NO MORE THAN 1000

EE-1986-D-P00-FIN

Dedicated to
all those who feel bad about me at
least once in their life time

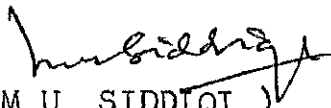
We say that the one and many become identified in our propositions and that now as in time past they run about together in every sentence which is uttered and that this union of them will never cease and is not now beginning but is as I believe an everlasting quality of propositions themselves which never grows old. But any young man when he first tastes these subtleties is delighted and fancies that he has found a treasure of wisdom in the first enthusiasm of his joy he leaves no stone or rather no thought unturned now rolling up the many into the one and kneading them together now unfolding and dividing them he puzzles himself first and above all and then he proceeds to puzzle his neighbors whether they are older or younger or of his own age - that makes no difference neither father nor mother does he spare no human being who has ears is safe from him hardly even his dog and a barbarian would have no chance of escaping him if an interpreter could only be found

Socrates (The dialogues of Plato
Jowett vol III Philebus p 566)

CERTIFICATE

Certified that this work FINITE DECONVOLUTION DATA
RECOVERY SCHEMES FOR BLOCK DATA COMMUNICATION OVER DISPERSIVE
CHANNELS by Mr P G Poonacha has been carried out under my
supervision and that this has not been submitted elsewhere for
a degree

October 1986


(M U SIDDIQI)
Department of Electrical Engineering
Indian Institute of Technology
Kanpur

ACKNOWLEDGEMENTS

In trying to understand my limitations in order to overcome them if possible I have spent very useful five and a half years at IIT Kanpur in the pleasant company of generous wise and critical teachers friends and wellwishers who have constantly encouraged me to grow and benefit from all possible sources To all of them I express my deep gratitude I feel indebted to this great institution for its free academic environment and to those faculty members who are striving hard to maintain that freedom intact

I feel happy to express my deepest sense of gratitude to my teacher and thesis advisor M U Siddiqi for giving me unconditional freedom to learn anything that I wished I am overwhelmed by his ability to understand and guide a guy of my nature and abnormality This thesis owes a great deal to his patience and critical remarks at all the stages of work He went through the entire manuscript many times in his inimitable way improving the contents and style in many places without losing sight of the overall objective of the thesis I am deeply moved by his genuine concern in my personal growth without expecting anything in return

I wish to express heartfelt thanks to my teachers P R K Rao V P Sinha S C Prabhu and K R Sarma for their affection timely advises many interesting and useful discussions and more than anything for allowing me to speak freely

without getting a feeling of inferiority complex. They have not only helped me a great deal in sharpening my professional skills but also played decisive roles in forming my values and shaping my conduct. My indebtedness to each one of them is more than words of thanks can ever convey.

Sundar Rajan, Govinda Rajan, Uday Madhu, Rajhu Shastri, Ashok have put themselves to a lot of trouble over helping me with this thesis. I acknowledge my indebtedness to them. I don't have words to thank them and a large number of friends who have made my stay at IIT Kanpur all these years a memorable pleasant experience.

It gives me great pleasure to thank my close friend C.M. Abraham for efficient neat and careful typing and his family for tolerating my visits during odd hours of the day. Thank you Abraham.

I also wish to thank Qasim Husain for neatly drawn figures, Tiwari and Gangaram for efficient cyclostyling work and Ahmed Binding Centre for neat binding work.

Finally, whatever little I have done so far owes more than anyone to my dear parents and sisters. I am eternally indebted to them for never valuing me in terms of marks, grades, position, money and name. Due to this attitude my home has always been a wonderful place for me to stay and enjoy. Thank you Appa and Amma for always affectionately refusing to lose arguments by saying, "God gives you energy to criticise God."

Poor Guy

TABLE OF CONTENTS

	LIST OF TABLES	x
	LIST OF FIGURES	xi
	LIST OF SYMBOLS AND ABBREVIATIONS	xiii
	SYNOPSIS	xvi
CHAPTER 1	INTRODUCTION	1
	1 1 Motivation	2
	1 2 Review of Literature	4
	1 2 1 Data Recovery Schemes for Continuous Serial Data Transmission	5
	1 2 2 Data Recovery Schemes for Block Data Transmission	8
	1 3 Key Results and Observations	9
	1 4 Organization of the Thesis	16
	1 5 Notations and Conventions	16
CHAPTER 2	BLOCK DATA TRANSMISSION	18
	2 1 Assumptions	18
	2 2 Discrete Time Model for ISI Channels	21
	2 3 Block Data Transmission Format	24
	2 4 The Deconvolution Problem	27
CHAPTER 3	MAXIMUM LIKELIHOOD DECONVOLUTION	32
	3 1 Recursive Algorithm for Minimization of Nonlinear Function of Several Variables	33
	3 1 1 Real Variable Case	38
	3 1 2 Discrete Variable Case	40

3 2	Recursive Algorithm for Banded Toeplitz Quadratic Functions	43
3 3	Viterbi Algorithm for Minimizing BTQF	46
3 4	Strategies for Reducing Computational and Storage Requirements	51
3 4 1	Binary Data Transmission	54
3 4 2	Minimization of Pseudo-Boolean Functions	56
3 5	Recursive Algorithm for Decoding Block Error Correcting Codes	59
CHAPTER 4	DECISION FEEDBACK DECONVOLUTION SCHEMES	70
4 1	Decision Feedback Scheme Using Cholesky Decomposition	72
4 2	MMS _E Decision Feedback Data Recovery Scheme	77
4 3	Decision Feedback Scheme Based on Minimum Property of Fourier Expansions	81
4 4	Decision Feedback Scheme Based on Modified Austin's Approach	87
4 4 1	Computation of the Tap Coefficients	93
4 4 2	Modified Austin's Approach and Nonlinear Gaussian Elimination Method	94
4 5	Some Observation	96
CHAPTER 5	DATA RECOVERY SCHEMES USING k-CIRCULANT COMPLETIONS	98
5 1	k-Circulant Completion and Finite Deconvolution	99
5 1 1	Deconvolution Using Circulant Completion	101
5 1 2	Deconvolution using k-Circulant Completion	101
5 1 3	Need for Deconvolution by k-Circulant Completion	104

5 2	Deconvolution by Direct Inversion	105
5 3	Data Recovery Schemes Based on k-Circulant Completions	111
5 4	Performance Considerations	113
CHAPTER 6	LEAST SQUARES DECONVOLUTION VIA k-CIRCULANT COMPLETIONS	121
6 1	Square Completion and Solution of Linear Equations	124
6 2	Least Squares Minimum Norm Solution and Moore-Penrose Generalized Inverse	126
6 3	Least Squares Solution Via Square Completions	127
6 4	Channels Amenable to LS Solution Via k-Circulant Completions	130
6 5	Characterization of Orthogonal k-Circulants	133
6 6	LS Solution Using k-Circulant Completion	136
CHAPTER 7	LEAST SQUARES DECONVOLUTION - ITERATIVE APPROACH	142
7 1	Review of Steepest Descent Algorithm	145
7 2	Application of DFT to Steepest Descent Algorithm	146
7 3	Derivation of the DFT Domain Steepest Descent Algorithm	148
7 3 1	Data Recovery Scheme Based on SDA in the DFT Domain	153
7 4	Rate of Convergence of SD	155
7 5	Fast Converging Steepest Descent Algorithm	156
7 6	Fixed Step Steepest Descent Algorithm	158

	Page
CHAPTER 8 PERFORMANCE OF DATA RECOVERY SCHEMES	164
8.1 Good Channels for Block Data Transmission	165
8.1.1 Maximum Minimum Distance Channels	165
8.1.2 Maximum Distance Channels	170
8.1.3 Procedure for Obtaining MMD and MD Channels	171
8.1.4 Computation of Minimum Distance of a Channel	174
8.2 Channels Used in the Study	176
8.3 Simulation Studies on Data Recovery Schemes	177
8.3.1 The Effect of Increased ISI among the Block Elements	191
CHAPTER 9 CONCLUSIONS AND SUGGESTION FOR FURTHER RESEARCH	197
APPENDIX A MITTEN'S LEMMA	203
APPENDIX B k -CIRCULANT MATRICES	204
REFERENCES	209

LIST OF TABLES

Table No	Title	Page
3 1	Stage 1	41
3 2	Stage 2	41
3 3	Stage 1	44
3 4	Stage 2	45
3 5	Stage 3	45
3 6	Stage 4	45
3 7	Stage 5	46
3 8	Stage 6	46
8 1	ome examples of vD and WID channels	173
8 2	Channels used in the simulation study	176
8 3	Chann l sample response coefficients	193

LIST OF FIGURES

Figure No	Title	Page
2 1	Block diagram of data transmission system	19
2 2	Discrete model of finite impulse response channel	22
2 3	Response of a channel to CSLT and BDT in the absence of noise	26
2 4	Plot of g versus m	28
3 1	Trellis diagram showing survivor paths	50
3 2	Development of the tables in ML-RDA for example 3 1	52
4 1	Block diagram of the data recovery scheme for BDT using Cholesky decomposition	75
4 2	Block diagram of the data recovery scheme using minimum property of Fourier expansion	85
4 3	Block diagram of the data recovery scheme using modified Austin's approach	91
5 1	Block diagram of the data recovery scheme using k-DFT	112
5 2	Data recovery scheme based on k-Circulant completion	112
7 1	Data recovery scheme based on DFT domain steepest descent algorithm	154
8 1	Comparison of performance of various DR schemes	179
8 2	Comparison of performance of various DR schemes	180
8 3	Comparison of performance of various DR schemes	181
8 4	Comparison of performance of various DR schemes	182
8 5	Comparison of performance of various DR schemes	183
8 6	Comparison of performance of various DR schemes	184
8 7	Comparison of performance of various DR schemes	185

Figure No	Title	Page
8 8	Comparison of performance of various DR schemes	186
8 9	Comparison of performance of various DR schemes	187
8 10	Comparison of MLB and VLC performance	189
8 11	MMD channel performance	191
8 12	Effect of m on the performance of CCB and SCB	192
8 13	Effect of increased ISI on the performance of MLB	194
8 14	Effect of increased ISI on the performance of MAB	194
8 15	Effect of increased ISI	196
8 16	Effect of increased ISI	196

LIST OF SYMBOLS AND ABBREVIATIONS

$\ A\ _e$	Euclidean norm of a matrix A
BDT	Block data transmission
BTQF	Banded Toeplitz quadratic function
CSDT	Continuous serial data transmission
DF	Decision feedback
DFE	Decision feedback equalizer
DFT	Discrete Fourier transform
DR	Data recovery
d_{\min}	Minimum distance
FFT	Fast Fourier transform
$F(\underline{x})$	Function of n variables where $\underline{x} = (x_1 \ x_2 \ \dots \ x_n)^T$
g	Number of ISI components
GE	Gaussian elimination
GF(2)	Galois field of two elements 0 and 1
H	Channel convolution matrix
H_k	k Circulant completion of H
$h_1 \ 0 \leq 1 \leq g$	Channel parameters
H^{LS}	Least squares inverse of H
$h(t)$	Channel impulse response
$H(z)$	z-transform of the autocorrelation function of $h(t)$
H_1	Circulant completion of H
H_{-1}	Skew-Circulant completion of H
H^+	Moore Penrose inverse of H

ISI	Intersymbol interference
LS	Least squares
MD	Maximum distance
MMD	Maximum minimum distance
ML	Maximum likelihood
ML-RDA	Maximum likelihood recursive deconvolution algorithm
MSE	Mean square error
N_0	Noise variance
PAM	Pulse amplitude modulation
PBF	Pseudo Boolean function
$(\text{PER})_1$	Probability of 1th data symbol being in error
P_e	Probability of error
$Q(x)$	$= \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-y^2/2} dy$
$r(A)$	Rank of a matrix A
SDA	Steepest descent algorithm
SNR	Signal-to-noise ratio
$\{S_k\}$	State sequence
T_c	Channel data rate in symbols per second
T_h	Channel impulse response duration
T_s	Source data rate in symbols per second
w_1	Sequence of independent Gaussian random variables with mean zero and variance $N_0/2$

VA	Vit rbi algorithm
$\{x_i\}$	Sequence of statistically independent data symbols
$X(z)$	z -transform of the autocorrelation function of $h(t)$
$y_k\}$	Output sequence
η_k	k -Circulant matrix with first row $(0 \ 0 \ 0 \ \dots \ k)$

SYNOPSIS

Poonacha P G Ph D
Department of Electrical Engineering
Indian Institute of Technology Kanpur
October 1986
FINITE DECONVOLUTION DATA RECOVERY SCHEMES FOR
BLOCK DATA COMMUNICATION OVER DISPERSIVE CHANNELS

In this thesis we study several data recovery (DR) schemes for block data transmission (BDT) over finite impulse response channels in the presence of noise. In block data transmission due to Clark the input sequence is partitioned into blocks which are separated by guard zones of appropriate duration such that the channel output blocks do not overlap. Since the channel is assumed to be linear its output can be represented as convolution of the input data values and the channel impulse response. The operation of recovering data from the nonoverlapping blocks at the receiver is then essentially that of finite deconvolution in the presence of noise.

We formulate the deconvolution problem as minimization of a banded Toeplitz quadratic function with data as unknown variables. When the unknown variables are real minimization of the quadratic function is straightforward. However when the variables take values from a finite alphabet as in the case of data transmission it is difficult to provide computationally attractive solutions. The solution to the deconvolution problem is

optimal or suboptimal depending on the assumptions made on the unknown variables and the strategy adopted for solving the problem

We study maximum likelihood decision feedback and least squares deconvolution methods in detail. Among these schemes maximum likelihood scheme is an optimal deconvolution scheme while decision feedback and least square deconvolution schemes are suboptimal. Our main concern throughout is to understand various theoretical and computational issues involved in developing DR scheme based on these methods.

Salient features of the DR schemes based on the above-mentioned deconvolution methods are briefly described in the following paragraphs

1 Maximum Likelihood Deconvolution Scheme

We first derive a maximum likelihood (ML) recursive deconvolution algorithm (ML-RDA) based on Mitten's lemma which provides conditions for optimal decomposition of multistage decision processes. This algorithm performs essentially the job of ML sequence estimation based on Viterbi algorithm. However, the derivation of the algorithm does not require the use of state representation and explicit invocation of Bellman's principle of optimality which are essential in deriving Viterbi algorithm. The approach taken provides better insight into the ML deconvolution problem compared to the conventional VA application. We have shown that if the data values are assumed to

be real variables the recursive algorithm reduces to the familiar Gaussian elimination algorithm for solving a set of linear equations. We can therefore view ML deconvolution algorithm as a generalized Gaussian elimination algorithm for minimizing banded Toeplitz quadratic objective functions.

In the case of binary data we show that the ML RDA reduces to a special case of the algorithm given by Hammer et al. for minimizing pseudo Boolean functions. We further show that in the binary case the problem of reducing storage and computational complexity of ML RDA involves determination of minimal sum of products form of Boolean functions of several variables. Therefore fast algorithm available for such purpose can be used to speed up computations and to reduce storage requirements. Finally we show that the recursive algorithm derived for minimizing pseudo Boolean functions can also be used for ML-decoding of binary block codes. Usefulness of the method is illustrated with the help of examples. However much remains to be done in this direction.

2 Decision Feedback Deconvolution Schemes

We next consider the possibility of incorporating decision feedback (DF) in DR schemes for BDT with the objective of achieving good performance without going in for ML-deconvolution which may not be economical in some practical situations.

DF-DR schemes consist of a forward filter for minimizing the effect of future symbols a nonlinear decision device and a backward filter for eliminating the effect of past symbols This scheme may be viewed as Gaussian elimination method of solving linear equations using forward elimination and backward substitution with a nonlinear decision device interposed between forward elimination and back substitution stages

For deriving DF-DR schemes we first minimize the banded Toeplitz quadratic function assuming that the data values are real This leads to the solution of banded Toeplitz system of linear equations We then derive three DF-DR schemes for BDT The first scheme is based on Cholesky decomposition of positive definite matrices This scheme however does not take noise variance into account The second DF-DR scheme takes noise variance into account and is based on minimum mean square error criterion The third DF-DR scheme is based on minimum property of Fourier expansions over finite dimensional Hilbert spaces This scheme is useful when the channel autocorrelation matrix is illconditioned and when no knowledge of the noise variance is available

Finally we derive a DF-DR scheme by modifying Austin's detection theory approach We show that the modified Austin's method can also be viewed as Gaussian elimination method with a nonlinear decision device

Least Squares Deconvolution Schemes-Direct Approach

We then undertake a critical study of least squares (LS) deconvolution using discrete Fourier transform (DFT) techniques. The main feature of our study is in providing a general treatment to the LS-deconvolution problem using the notion of square completions of the channel convolution matrix. We show that both LS solution and the conventional DFT approach are particular cases of square completions of the channel convolution matrix. Moreover, DFT approach to the deconvolution problem is a particular case of k -Circulant completion of the channel convolution matrix. Based on k -Circulant completions we develop two data recovery schemes which are easily implementable and have reduced computational complexity.

Although DFT based deconvolution is widely used, conditions under which such techniques provide optimal solution to deconvolution problem according to some criterion do not seem to have been explicitly mentioned. We investigate conditions on k -Circulant completions which provide LS solution to the deconvolution problem. Using the framework of square completion of matrices we obtain conditions on square completion that provide Moore-Penrose inverse of a full rank matrix. Such completions are called orthogonal completions. Then it is shown that orthogonal k -Circulant completions are orthogonal k -Circulant matrices. Using this result we show that for all practical

cases of interest LS solution to the deconvolution problem cannot be obtained via direct inversion of k Circulant completions

4 Least Squares Deconvolution Scheme - Iterative Approach

Finally we consider an iterative deconvolution scheme using Steepest Descent Algorithm (SDA). One of the purposes of this study is to examine the observation of several authors working with fixed step SDA in the DFT domain that the DFT domain iteration scheme converges faster than the sample domain scheme. For this purpose we derive SDA in the DFT domain from three different viewpoints. We show that the DFT domain scheme does not alter the convergence rate. It only helps in reducing computational complexity at each iteration. However, it is shown that k -Circulant completions can be used to obtain good estimates of the convergence parameter in the case of some examples. These estimated values can be used either in the sample domain or in the DFT domain to accelerate convergence.

Since the error performance of each data recovery scheme strongly depends on the particular channel used, we give an account of good channels for the purpose of comparing the performance of various DR schemes. Two criteria for good channels are considered. The first criterion provides channels having maximum minimum distance (d_{min}) called d_{min} -channels. These channels are used as benchmarks for comparing the performance of various DR schemes used on actual channels.

Determination of MMD-channels however is a difficult problem. Therefore we consider the second class of good channels called maximum distance channels (MD-channels) which maximize the total distance. A procedure is given for obtaining MMD-channels via MD-channel characterization. For the purpose of comparing actual channels with MMD-channels we also develop a recursive algorithm for computing the minimum distance of a given channel. It is shown that some of the MMD and MD channels are partial response channels.

The performance of some of the data recovery schemes discussed earlier has been evaluated using computer simulation. Performance curves showing the variation of probability of error with signal-to-noise ratio are given for a wide range of block lengths and typical channel examples.

The simulation study brings out the following merits of BDT over continuous serial data transmission (CSDT):

- 1) If decision feedback idea is used to improve the performance of BDT systems, error propagation is limited to almost a block length in the worst case, whereas in the case of DF-DR schemes for CSDT, error propagation can extend over much longer duration. Since in the range of medium signal-to-noise ratios the error propagation is a serious problem with DF-DR schemes for CSDT systems, BDT with DF-DR schemes are to be preferred in such applications.

2) With an appropriate choice of the block length BDT can be used to provide better trade-off between performance processing complexity and data rate

A drawback of BDT is that the effective data rate over the channel is less compared to CSDT for a given source rate. The loss in data rate may be compensated by transmitting data over the channel at a higher rate. This results in increased intersymbol interference among the block samples. Our simulation studies however indicate that if the block length is properly chosen the degradation in performance of DR schemes for BDT with increased intersymbol interference will not be significant.

CHAPTER 1

INTRODUCTION

In this chapter we study several data recovery (DR) schemes for block data transmission over linear time dispersive channels in the presence of noise. In block data transmission (BDT) the input data sequence is partitioned into blocks which are separated by guard zones of appropriate duration such that the channel output blocks do not overlap. Since the channel is assumed to be linear its outputs can be represented as convolution of the input data values and the channel impulse response. The operation of recovering data from the nonoverlapping blocks at the channel output is then essentially that of finite deconvolution in the presence of noise. That is, given the channel impulse response and the nonoverlapping channel output blocks which have been corrupted by noise, the problem is to recover the data at the receiver end in some optimum manner.

We formulate the deconvolution problem as minimization of a banded Toeplitz quadratic function with data as unknown variables. When the unknown variables are real minimization of the quadratic function is straightforward. However, when the variables take values from a finite alphabet as in the case of data transmission, it is difficult to provide computationally attractive solutions. The solution to the deconvolution problem is

optimal or suboptimal depending on the assumptions made on the unknown variables and the strategy adopted for solving the problem

We study maximum likelihood decision feedback and least squares deconvolution methods in detail. Among these schemes maximum likelihood (ML) scheme is an optimal deconvolution scheme while decision feedback (DF) and least squares (LS) deconvolution schemes are suboptimal. Our main concern throughout is to understand various theoretical and computational issues involved in developing DR schemes based on these methods

1.1 MOTIVATION

Design and evaluation of DR schemes for long strings of data transmitted over time dispersive channels in the presence of noise is an important topic of study in the area of digital communication. Development of various data transmission and recovery schemes which date back to the pioneering work of Nyquist [1] and which continue to draw a great deal of attention even today has been primarily motivated by a desire to increase the data rate. This is reflected in the design of most of the available data transmission and recovery schemes. In this connection continuous serial data transmission (CSDT) happens to be the most widely used data transmission scheme in which a continuous string of serial data is transmitted over the channel.

In the case of data transmission over time dispersive channels due to inter symbol interference (ISI) the signal samples at the receiver input become highly correlated. Therefore implementation of optimal DR schemes for CSDT system is non practical as they require very large observation intervals which means prohibitively large delays and processing requirements. One method of circumventing these problems which has received considerable attention is to make decisions based on truncated observation intervals. This strategy reduces delay and processing time at the expense of increased probability of error. The trade-off however appears to be one sided from the point of view of the following three major demands a good data transmission system is expected to satisfy

D1 Minimum probability of error The probability of a detected data symbol being in error must be minimum

D2 Minimum processing time Computational complexity of the data recovery scheme must be minimum and the scheme must be easily implementable

D3 Maximum data rate It must be possible to transmit data values one after another with minimum time gap between adjacent symbols

The foregoing discussion highlights the fact that there is still need for investigating alternate ways of data transmission which will help in striking a better compromise between D1 D2 and D3

An alternate method of reducing delay and processing requirement without sacrificing error performance is to reduce the observation interval by appropriately changing the data transmission format. This may be achieved by using BDT format studied by Clark [2] - [4]. We study various data recovery schemes for BDT with the following objectives in mind

- 1 A better understanding of the possible trade-off between performance, processing complexity and data transmission rate in the context of BDT
- 2 A better appreciation of the merit and demerits of the existing DR schemes for CSDT
- 3 To critically analyse the advantages of BDT in situations where probability of error performance of the transmission system is more important than achieving maximum data rate at the cost of error performance as in the case of CSDT

1.2 REVIEW OF LITERATURE

To place our study in a proper perspective we shall first review the existing literature on DR schemes for CSDT and BDT. Our interest is not to give a detailed account of the historical development of various DR schemes. Instead we wish to highlight some of the key development with the help of relevant literature. It may be pointed at the outset that most of the work is in connection with CSDT. References on BDT are very few. We review the literature on CSDT first and consider BDT later.

An alternate method of reducing delay and processing requirements without sacrificing error performance is to reduce the observation interval by appropriately changing the data transmission format. This may be achieved by using BDT format studied by Clark [2] - [4]. We study various data recovery schemes for BDT with the following objectives in mind

- 1 A better understanding of the possible trade-offs between performance, processing complexity, and data transmission rate in the context of BDT
- 2 A better appreciation of the merit and demerits of the existing DR schemes for CSDT
- 3 To critically analyze the advantages of BDT in situations where probability of error performance of the transmission system is more important than achieving maximum data rate at the cost of error performance as in the case of CSDT

1.2 REVIEW OF LITERATURE

To place our study in a proper perspective we shall first review the existing literature on DR schemes for CSDT and BDT. Our interest is not to give a detailed account of the historical development of various DR schemes. Instead we wish to highlight some of the key developments with the help of relevant literature. It may be pointed at the outset that most of the work is in connection with CSDT. References on BDT are very few. We review the literature on CSDT first and consider BDT later.

1 2 1 Data Recovery Schemes for Continuous Serial Data Transmission

The literature on DR schemes for CSDT in the presence of ISI and noise is rich with a wide variety of novel approaches and related theoretical developments. A number of well written books are available on this topic. Texts by Bennet and Davey [5], Lucky, Salz and Widom [6] and Proakis [7] provide excellent coverage of most of the important topics in the area of CSDT along with long lists of references. The text book by Clark [4] provides wide coverage to both CSDT and BDT. The collection of papers edited by Franks [8] on data communication provides at a place twenty four important works on CSDT along with informative and well written long editorial comments. Important developments that took place in the domain of CSDT during 1968-73 are discussed with extensive bibliography in a survey article by Lucky [9]. Most of the reported data recovery schemes for CSDT are either some variant of one of the following three schemes or their combinations.

1 Linear Data Recovery Schemes

The first and the most simple one is the linear DR scheme consisting of a matched filter matched to the channel followed by a transversal filter whose output is sampled and used in making decisions. First systematic study in this connection appears to be due to Tufts [10] [11]. Extending the classical

work of Nyquist [1] Tufts [10] has derived an optimal linear receiver structure using minimum mean square error (MMSE) criterion. As an extension of this work Smith [12] has obtained explicit solution to the optimum receiver when the transmitted waveform is subjected to an average power constraint. The criterion of optimization however remains same as before.

Deviating considerably from the previous approaches and using detection theory point of view Austin [13] has shown that linear receiver can be derived on the assumption that the data values are independent Gaussian random variables. The resulting transversal filter in general is of infinite length. However for practical purposes only finite length filters are employed and the tap coefficients are optimized by minimizing the MSE between the filter output at the sampling instant and the actual data value. Due to their simplicity finite transversal filters have been used extensively for the purpose of automatic channel equalization [14]-[22] as well as for adaptive equalization [23]-[28].

2 Decision Feedback Data Recovery Schemes

The second DR scheme known as decision feedback equalizer (DFE) [13] [29] consists of a matched filter sampler forward filter backward filter and a nonlinear decision device. DFE makes use of past decisions to improve performance. However as the actual decisions are always prone to error (except possibly

at high signal-to-noise ratio (SNR)) DFE suffers from the problem of error propagation. DFE structure can be derived on the assumption that the past symbols are error free and that the future symbols are independent Gaussian random variables. Using this assumption Austin [13] has derived the structure of the DFE. The resulting forward filter is in general of infinite length. However, as in the case of linear DR schemes, it is common practice to use only finite length and the tap coefficients are optimized using MSE criterion. DFE has since been studied from various viewpoints [30]-[37].

3 Maximum Likelihood Data Recovery Scheme

The third DR scheme known as maximum likelihood DR scheme is due to Forney [38]. He employed Viterbi algorithm (VA) [39]-[42] to alleviate the computational complexity associated with ML sequence estimation of data. The ML sequence estimator minimizes the probability of sequence error when all the sequences are equally likely. We wish to note here that an algorithm similar to VA has been developed by Kobayashi [43] for ML decoding of a particular type of PAM signal before the publication of the Forney's now classical work [38].

For ML sequence estimation the VA requires entire observation values. For long data streams such a scheme therefore becomes unattractive both from the point of view of computational complexity and storage requirements. To make the VA

useful decisions are made after a fixed number of observations becomes available. This number is fixed empirically on the basis of channel memory with the assumption that degradation due to such premature decision is negligible [30]. However, since the performance degradation strongly depends on the channel and noise such an assumption needs careful examination.

After Forney's work all efforts on ML DR schemes seem to have been directed towards further reducing the complexity of VA [4]-[52]. Specifically, since the computational complexity of the VA grows exponentially with the channel memory, efforts have been made to shorten the channel impulse response duration by using prefilters.

1.2.2 Data Recovery schemes for Block Data Transmission

Clarke [2] [4] has investigated a number of adaptive detection processes for BDT systems. In his work the BDT has been referred to as transmission of orthogonal group of signal elements by virtue of the fact that at the channel output such groups do not overlap. The strategy used by him is based on the recognition that the detection of elements in a group can be carried out by solving a set of linear simultaneous equations using iterative techniques. In particular, the adaptive detection processes studied by him make use of the point Gauss-Seidel iteration method. He has also considered several modifications of this basic detection scheme to take care of the

discrete nature of the data. From the extensive simulation work carried out by him and his group [4] it appears that such modifications lead to efficient and cost effective DR schemes whose performance is comparable to ML approach. Moreover such modifications do not require excessive number of sequential operations. Extending this approach Clark and Ghani [53] have studied detection processes for multilevel signalling schemes. Simulation results presented by them show that good tolerance to noise can be achieved by such detection processes which do not require excessive number of sequential operations.

Before winding up this section we wish to take note of the work of Klein and Wolf [54] in which BDT is considered from a different point of view. They have investigated certain aspects of the possibility of designing a receiver filter such that the overall response of the channel and the receiver filter resembles a cyclic code generator. However this line of thought does not seem to have been pursued further.

1.3 KEY RESULTS AND OBSERVATION

As mentioned earlier the basic approach in our study is to view the data recovery problem as a finite deconvolution problem in the presence of noise. Salient features of the study are briefly described in the following paragraphs.

1 Maximum Likelihood Deconvolution Scheme

We first derive a ML recursive deconvolution algorithm (ML-RDA) based on Mitter's lemma [55] which provides conditions for optimal decomposition of multistage decision processes. This algorithm performs essentially the job of ML sequence estimation based on VA. However, the derivation of the algorithm does not require the use of state representation and explicit invocation of Bellman's principle of optimality [56] which are essential in deriving VA. The approach taken provides better insight into the ML deconvolution problem compared to the conventional VA application. We have shown that if the data values are assumed to be real variables the recursive algorithm reduces to the familiar Gaussian elimination algorithm for solving a set of linear equations. We can therefore view ML-deconvolution algorithm as a generalized Gaussian elimination algorithm for minimizing banded Toeplitz quadratic objective functions.

In the case of binary data we show that the ML-RDA reduces to a special case of the algorithm given by Hammer et al. [57] for minimizing pseudo Boolean function. We further show that in the binary case the problem of reducing storage and computational complexity of ML-RDA involves determination of minimal sum of products form of Boolean functions of several variables. Therefore fast algorithms available for such purpose can be

used to speed up computations and to reduce storage requirements. Finally we show that the recursive algorithm derived for minimizing pseudo Boolean functions can also be used for ML-decoding of binary block codes. Usefulness of the method is illustrated with the help of examples. However much remains to be done in this direction.

2 Decision Feedback Deconvolution Schemes

We next consider the possibility of incorporating decision feedback in DP schemes for BDT with the objective of achieving good performance without going in for ML-deconvolution which may not be economical in some practical situation.

DF-DR schemes consist of a forward filter for minimizing the effect of future symbols, a nonlinear decision device and a backward filter for eliminating the effect of past symbols. This scheme may be viewed as Gaussian elimination method of solving linear equations using forward elimination and backward substitution with a nonlinear decision device interposed between forward elimination and back substitution stages.

For deriving DF-DR schemes we first minimize the banded Toeplitz quadratic function assuming that the data values are real. This leads to the solution of banded Toeplitz system of linear equations. We then derive three DF-DR schemes for BDT. The first scheme is based on Cholesky decomposition of positive definite matrices. This scheme however does not take noise

variance into account. The second DF-DF scheme takes noise variance into account and is based on minimum mean square error criterion. The third DF-DF scheme is based on minimum property of Fourier expansions over finite dimensional Hilbert spaces [58]. This scheme is useful when the channel autocorrelation matrix is illconditioned and when no knowledge of the noise variance is available.

Finally, we derive a DF-DP scheme by modifying Austin's decision theory approach [15]. We show that the modified Austin's method can also be viewed as Gaussian elimination method with a nonlinear decision device.

3 Least Squares Deconvolution Schemes--Direct Approach

We then undertake a critical study of least squares deconvolution using discrete Fourier transform (DFT) techniques. The main feature of our study is in providing a general treatment to the LS-deconvolution problem using the notion of square completions of the channel convolution matrix. We show that both LS solution and the conventional DFT approach are particular cases of square completions of the channel convolution matrix. Moreover, DFT approach to the deconvolution problem is a particular case of k -Circulant completion of the channel convolution matrix. Based on k -Circulant completions, we develop two data recovery schemes which are easily implementable and have reduced computational complexity.

Although DFT based deconvolution is widely used [59]-[60] conditions under which such techniques provide optimal solution to deconvolution problem according to some criterion do not seem to have been explicitly mentioned. We investigate conditions on k -Circulant completions which provide LS solution to the deconvolution problem. Using the framework of square completion of matrices we obtain conditions on square completions that provide Moore-Penrose inverses of a full rank matrix. Such completions are called orthogonal completions. Then it is shown that orthogonal l -Circulant completions or orthogonal l -Circulant matrices. Using this result we show that for all practical cases of interest LS solution to the deconvolution problem can not be obtained via direct inversion of l -Circulant completion.

4 Least Squares Deconvolution Scheme-Iterative Approach

Finally we consider an iterative deconvolution scheme using steepest descent algorithm (SDA). One of the purposes of this study is to examine the observation of several authors [21] [22] [64] [66] working with fixed step SDA in the DFT domain that the DFT domain iteration scheme converges faster than the sample domain scheme. For this purpose we derive SDA in the DFT domain from three different viewpoints. We show that the DFT domain scheme does not alter the convergence rate but only helps in reducing computational complexity at each

iteration. However, it is shown that k-Circulant completions can be used to obtain good estimates of the convergence parameter in the case of some examples. These estimated values can be used either in the sample domain or in the DFT domain to accelerate convergence.

Since the error performance of each DR scheme strongly depends on the particular channel used, we give an account of good channels for the purpose of comparing the performance of various DR schemes. Two criteria for good channels are considered. The first criterion provides a channel having maximum minimum distance (MMD) called MMD channels. These channels are used as benchmarks for comparing the performance of various DR schemes used on actual channels.

Determination of MMD-channels, however, is a difficult problem. Therefore, we consider the second class of good channels called maximum distance channels (MD channel) which maximize the total distance. A procedure is given for obtaining MMD-channels via MD-channel characterization. For the purpose of comparing actual channels with MMD-channels, we also develop a recursive algorithm for computing the minimum distance of a given channel. It is shown that some of the MMD and MD channels are partial response channels.

The performance of some of the data recovery schemes discussed earlier has been evaluated using computer simulation.

Performance curves showing the variation of probability of error with signal-to-noise ratio are given for a wide range of block lengths and typical channel examples

The simulation study brings out the following merits of BDT over CSDT

1) If decision feedback code is used to improve the performance of BDT systems error propagation is limited to almost a block length in the worst case whereas in the case of DF-DR schemes for CSDT error propagation can extend over much longer duration. Since in the range of medium signal-to-noise ratios the error propagation is a serious problem with DF-DR schemes for CSDT system BDT with DF-DR schemes are to be preferred in such applications

2) With an appropriate choice of the block length BDT can be used to provide better trade-off between performance, processing complexity and data rate

A drawback of BDT is that the effective data rate over the channel is less compared to CSDT for a given source rate. The loss in data rate may be compensated by transmitting data over the channel at a higher rate. This results in increased ISI among the block samples. Our simulation studies however indicate that if the block length is properly chosen the degradation in performance of DR scheme for BDT with increased ISI will not be significant.

1.4 ORGANIZATION OF THE THESIS

Besides this chapter the thesis consists of 8 chapters. Chapter 2 gives a detailed account of the assumption made in the study channel model considered. A discussion of the BDT format and problem formulation. In Chapter 3 we develop a recursive algorithm for maximum likelihood deconvolution of data and study the application of this methodology for decoding or error correcting binary block codes. Decision feedback DR schemes are studied in Chapter 4. Application of k -Circulant matrices for designing cost effective DR schemes is studied in Chapter 5. In Chapter 6 we discuss conditions on k -Circulants for providing least squares solution to the data recovery problem. Application of steepest descent algorithm for LS-deconvolution is discussed in Chapter 7. In Chapter 8 we study the performance of some of the DR schemes developed in the thesis. Chapter 9 gives conclusions and suggestions for further research.

1.5 NOTATIONS AND CONVENTIONS

In this section we wish to mention some of the notations and conventions which are common to all the chapters of the thesis. Rest of the notations and conventions are explained in a list included after the list of figures. Lower case letters with a bar below are used to denote sample domain or time domain vectors. For example $\underline{\bar{a}}, \underline{\bar{h}}, \underline{\bar{y}}$ are sample domain vectors.

Upper case letters with a bar below represent DFT or frequency domain vectors of the corresponding sample or time domain vectors. Upper case letters without a bar below except R and C represent matrices. I_m and $O_{m \times n}$ respectively denote identity matrix of order m and null matrix of order $m \times n$. R^m and C^m denote m -dimensional real and complex vector spaces respectively. We use $R(A)$ to denote the range space of A - the space spanned by the columns of A and $N(A)$ to denote the null space of A - the space of all vectors \underline{x} such that $A\underline{x} = \underline{0}$. Lower case letters without under bar are used to denote variable and constants. Transpose of a matrix B is denoted as B^T . If B is a complex matrix B denote complex conjugate transpose of B . Similarly B^* denotes complex conjugate transpose of vector \underline{B} .

$|| \underline{x} ||$ denotes the Euclidean norm of the vector \underline{x} given by

$$|| \underline{x} || = (\underline{x}^* \underline{x})^{1/2}$$

$|| A ||$ denotes norm of matrix A . When A is symmetric $|| A || = \max_{1 \leq i \leq n} |\lambda_i|$ where $|\lambda_i|$ is the absolute value of the i th eigenvalue of $n \times n$ matrix A .

End of proof of theorems is indicated by double star (**). Computational complexity of algorithm is denoted by $O(g(n))$ which is to be read as order of $g(n)$.

CHAPTER 2

BLOCK DATA TRANSMISSION

In this chapter we set the stage for the development of later chapters. The assumptions made in the study are given in Section 2.1. Discrete model of the channel used in the study is discussed in Section 2.2. Block data transmission format and some of its details are given in Section 2.3. Finally in Section 2.4 we show that the deconvolution problem is equivalent to minimization of positive definite Toeplitz quadratic function with data as unknowns.

2.1 ASSUMPTIONS

Consider the data transmission system shown in Figure 2.1. In the figure input data stream to the transmitter filter is a sequence of impulses $\{x_i \delta(t - iT_s)\}$ where T_s is the intersymbol duration and $\delta(t)$ denotes an unit impulse function. x_i denotes the value of the i th transmitted symbol. The transmission path is a linear baseband channel consisting of a modulator bandpass channel and a demodulator. Noise is added at the output of the channel.

For the purpose of study we make following assumptions regarding the data transmission system

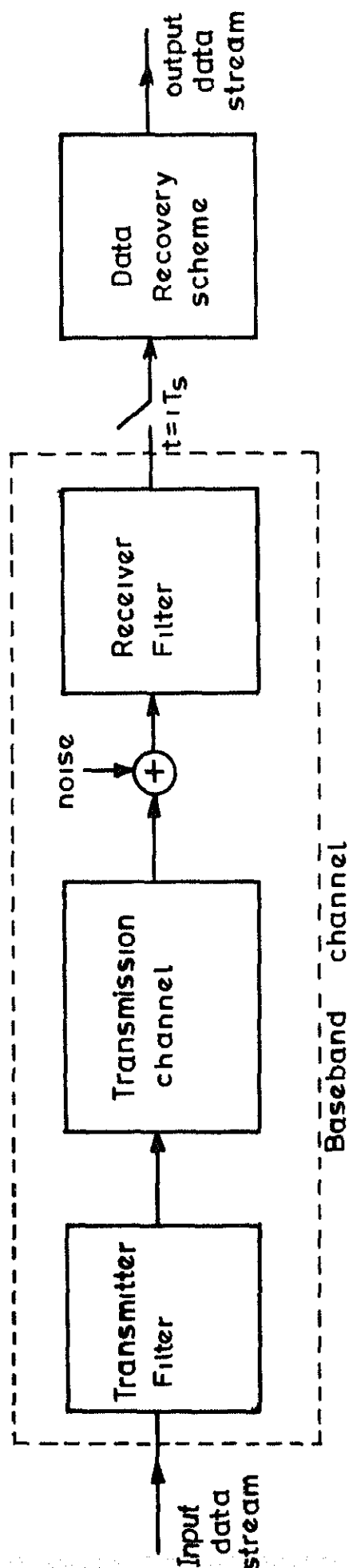


FIG 2 1 Block diagram of data transmission system

1 A synchronous data transmission system is assumed. That is the receiver has knowledge of the correct sampling instants.

2 The baseband channel is linear and is characterized by an impulse response $h(t)$. In most of the practical cases channels can be modelled either as time invariant or as slowly time varying. In the case of time varying channel data recovery schemes must be made adaptive. However, we will not study this aspect in this thesis.

3 The channel impulse response is of finite duration. This assumption is met by most of the practical channels. Although for a bandlimited channel the impulse response is of infinite duration in most of the cases impulse response becomes negligible after a finite duration.

4 Data symbols are statistically independent and take on values with equal probability from the set $\{\pm 1, \pm 3, \pm 5, \dots, \pm(q-1)\}$ where q is an even integer. This assumption is rarely met by any practical communication system. However, it can be realized to some extent with the help of scramblers at the transmitter end. The effect of scrambling can be removed by descrambling the data symbols at the receiver end (see Clark [4] page 178).

5 Noise samples are statistically independent and each sample is a Gaussian random variable with zero mean and variance $N_0/2$. The white Gaussian noise assumption is widely used as it considerably simplifies mathematical analysis and provides worst case performance.

It may be noted that although some of these assumptions may not be valid in practice they provide a useful framework for formulating and analysing the problem

2.2 DISCRETE TIME MODEL FOR ISI CHANNELS

Due to the presence of the sampler after the receiving filter (see Figure 2.1) it is convenient to analyse the base band data transmission system with a discrete time equivalent model. When a finite number of data symbols are transmitted over finite ISI channels in the presence of white Gaussian noise it can be shown [33] [7] that the channel model shown in Figure 2.2 provides an equivalent model for the estimation of the input sequence without any loss of error performance. For this to happen transfer function of the transversal filter $l(z)$ must be chosen such that

$$X(z) = H(z) H^*(z) \quad (2.1)$$

where $X(z)$ is the z-transform of the autocorrelation function of the channel impulse response $h(t)$ and $H(z)$ is a polynomial having roots of $X(z)$ which are inside the unit circle

Clark [3] has considered channel responses derived in a different manner. He has shown that effective multipath distortion introduced by the transmission path can be modelled as an equivalent sampled impulse response of the linear transmission path. The overall sampled response of the channel is obtained

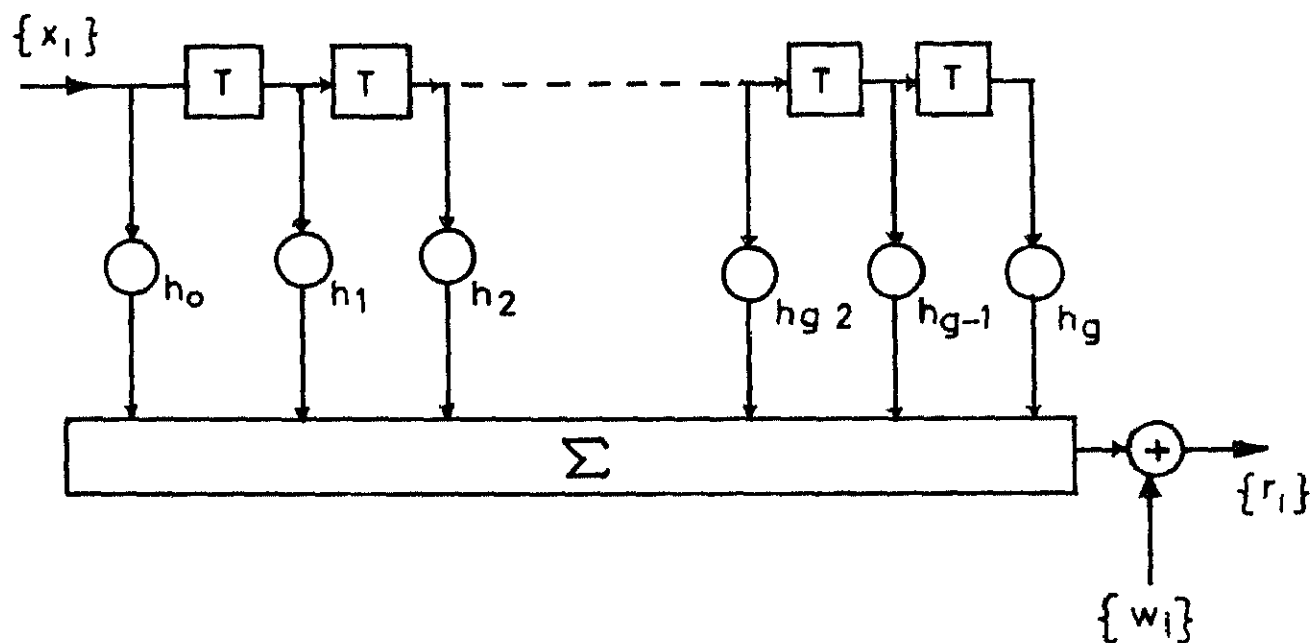


FIG 2 2 Discrete model of finite impulse response channel

by considering identical transfer functions for both transmitter and receiver filters

For the purpose of study we use the discrete time model for finite ISI channels shown in Figure 2. The model consists of a finite length transversal filter whose tap coefficients reflect the effect of transmitter filter, channel and receiving filter. Flexibility in the choice of the tap coefficients to take care of a wide class of channels including channel variations (by making the tap coefficients vary with time) and the simple manner in which it captures the essential idea of ISI and noise are the major considerations for using the model in the study.

In Figure 2.2 $\{x_1\}$ denotes a sequence of statistically independent data symbols $\{w_1\}$ denotes a sequence of independent Gaussian random variables with mean zero and variance $N_0/2$. $h_{1-1} = 0.1$ g are the channel parameters g indicates the number of ISI components and it depends upon the data rate and duration of the channel impulse response. As the data rate increases g also increases.

From the channel model of Figure 2.2 it is easy to see that the output sequence $\{r_1\}$ has the following representation

$$r_1 = \sum_{k=0}^g h_k x_{1-k} + w_1 \quad 1 = 0, 1, 2 \quad (2.2)$$

Data recovery problem in the case of CSDT systems is essentially that of solving the set of equations given by (2.2) for x_1 's. In the case of long strings of data this becomes a difficult problem due to the large number of equations which are to be solved in the presence of noise. Block data transmission format which overcomes this problem is described in the following section.

2.3 BLOCK DATA TRANSMISSION FORMAT

In BDT blocks of data of length m with each block followed by g zeros is transmitted through the channel. If the source data rate is $1/T_s$ symbols per second due to the g zero following each block the effective rate of transmission will be $m/(nT_s)$ symbols per second where $n = m + g$. On the other hand no reduction in data rate is experienced if the data symbols are transmitted through the channel at a rate of $1/T_c$ symbols per second where T_c is obtained as follows. Let g be chosen such that the channel dispersion duration (T_h) does not exceed $(g+1)T_c$ seconds. That is

$$(g+1)T_c \geq T_h \quad (2.3)$$

Then T_c is given by

$$T_c = nT_s/(m+g) \quad (2.4)$$

Substituting the value of T_c from (2.4) in (2.3) we obtain

$$g > m(T_h - T_s)/(mT_s - T_h), \quad mT_s > T_h \quad (2.5)$$

Due to the presence of g zeros following each transmitted block the channel is forced into zero state before the arrival of a fresh block at the input. Thus the effect of the channel is confined to blocks of length n where $n = m+g$. Therefore ISI does not propagate from one block to the subsequent ones. Due to this feature blocks can be processed independently at the receiver to recover data value. Note that two buffers will be needed of which one will be receiving the next block while the other holds the current block for processing.

Figure 2.3 helps to make some of the ideas expressed in the previous paragraphs more clear. In Figure 2.3a a rather simplistic channel is shown for the purpose of illustration. The data duration is assumed to be one third of the duration of the impulse response. Output of the channel shown in Figure 2.3a to an arbitrarily chosen string of binary (± 1) data is shown in Figure 2.3b. Figure 2.3c shows the corresponding overlapping pulse responses. Figures 2.3d and 2.3e show the corresponding situation in the case of BDT with increased data rate over the channel. In this case $m = 6$, $g = 4$ and $n = 10$. Moreover we obtain $T_c = 0.6 T_s$ as the data transmission rate over the channel. From the Figures 2.3d and 2.3e it can be seen that the blocks get separated at the channel output.

Certain features of the inequality (2.5) can be made clear with a graphical representation. For this purpose

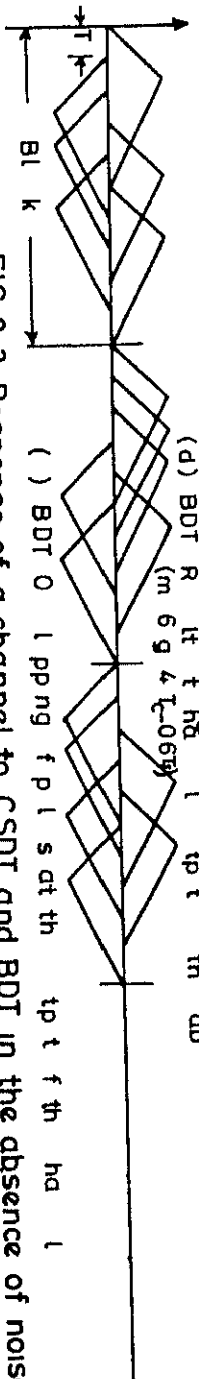
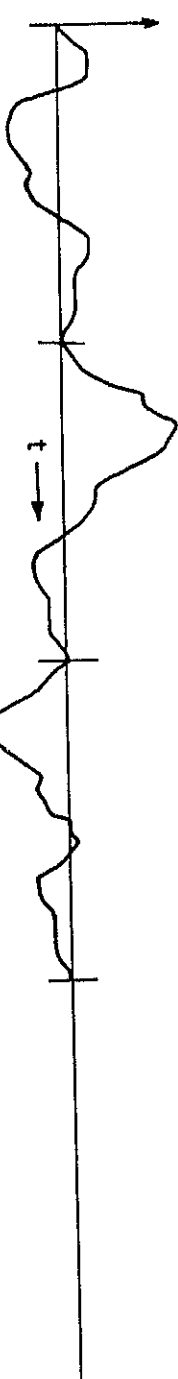
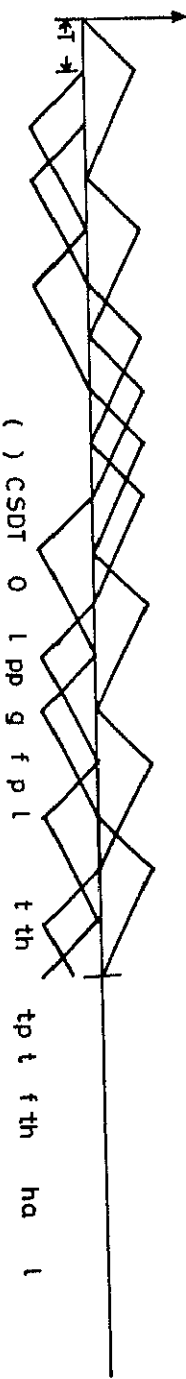
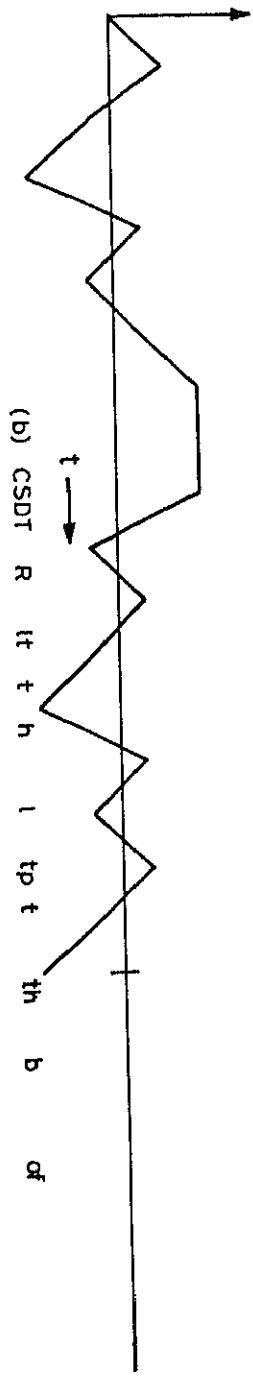
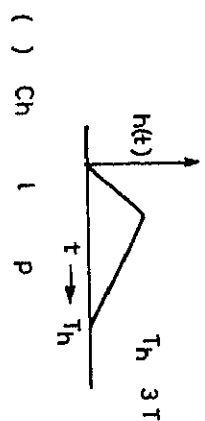


FIG 2 3 Response of a channel to CSDT and BDT in the absence of noise

Let $p = T_h/T_s$. Then (2.5) becomes

$$g > (p-1) m/(m-p) \quad m > p \quad (2.6)$$

In Figure 2.4 g is plotted against m for various values of p when g is given by

$$g = (p-1) m/(m-p) \quad m > p$$

Note that although n and g are integers in Figure 2.4 in order to avoid too many symbols g is plotted as a continuous function of m for $m > p$. From the figure it is clear that for a given p , m must be sufficiently large for g to be small. Under this condition T_c will be close to T_s . We also observe that the required value of g levels off fast after dropping rapidly in the beginning. This means that the effect of ISI among the members of a block will not become too large if moderate value of n is chosen.

2.4 THE DECONVOLUTION PROBLEM

In this section we formulate the deconvolution problem. Let h be an n -length vector with first g elements as the channel parameters and rest of the elements zero. Let x denote an m -length data vector and r a n -length observation vector. In terms of these vectors the convolution relationship given by (2.2) can be written as

$$\underline{r} = H \underline{x} + \underline{v} \quad (2.7)$$

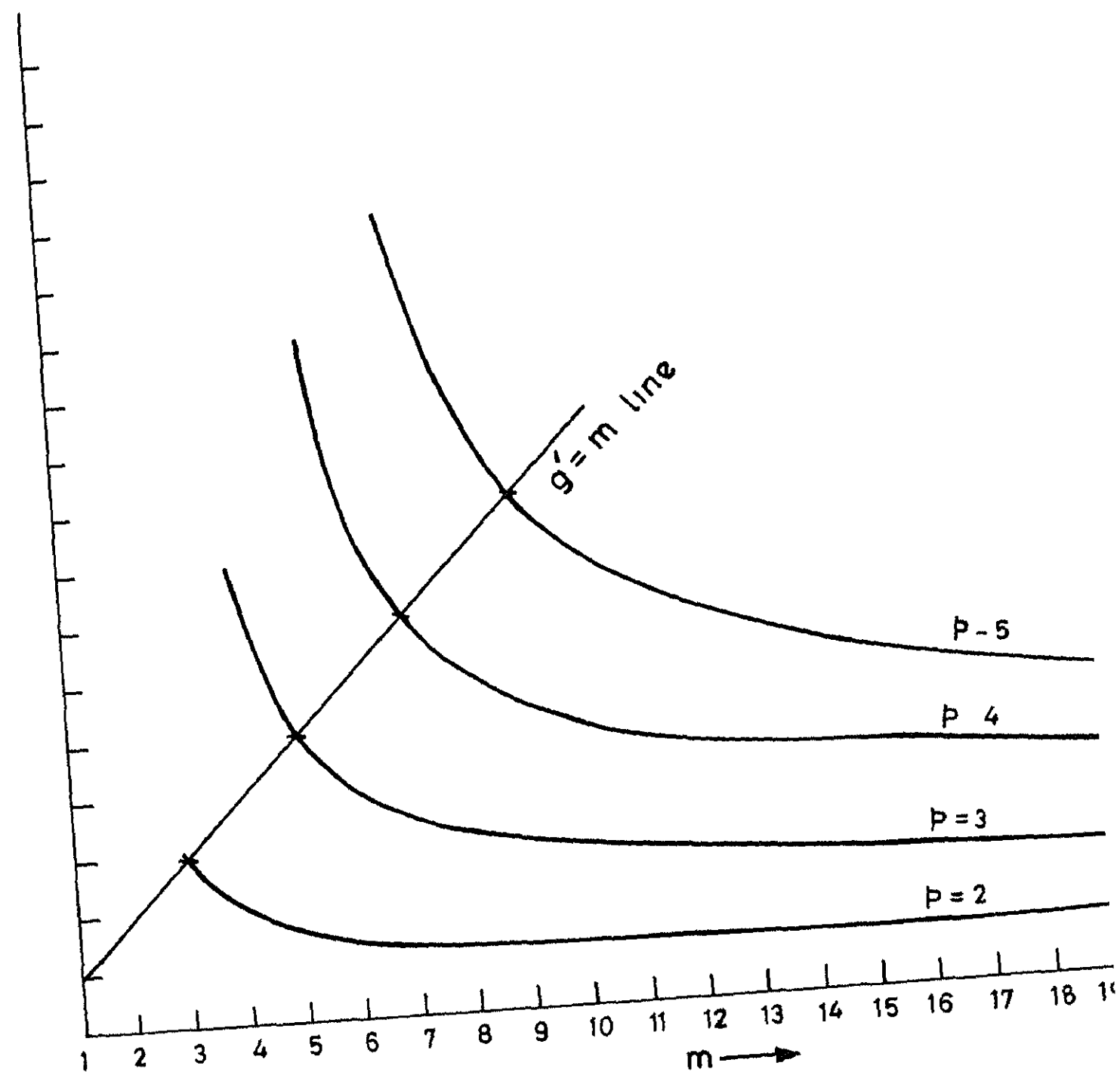


FIG 2 4 Plot of g' versus m

where H is the convolution matrix with h as its first column. For the same illustration H matrix is shown below

$$H = \begin{bmatrix} h_0 & 0 & 0 & 0 \\ h_1 & h_0 & 0 & \\ & h_1 & h_0 & \\ & & h_1 & 0 \\ h_g & & & h_0 \\ 0 & h_g & & \\ & 0 & h_g & \\ 0 & 0 & 0 & n_g \end{bmatrix}$$

the (i, j) th element of H is given by

$$h_{ij} = \begin{cases} h_{|i-j|} & i \leq j \\ 0 & \text{for } i > j \text{ and } i-j > g \end{cases}$$

$$i = 1, 2, 3, \dots, n \quad j = 1, 2, 3, \dots, m$$

The problem of determining x given r and h or finding h given r and x is called finite d convolution problem in the presence of noise.

The problem has received as much attention as the finite deconvolution problem due to its appearance in a wide variety of applications [59]-[61] [67]-[71]. However, most of the methods available for dealing with the problem assume that the unknown variables are real. Whereas in the context of BDT the data vector x take discrete values from a finite alphabet. Therefore in this case, we have to deal with a combinatorial deconvolution problem in which the space of all possible solutions do not possess the nice properties of R^n to which real vectors belong. One straightforward method of deconvolution is to search through all possible solutions. However, even in the binary case searching through all possible 2^m vectors becomes out of hand for moderate values of m . Moreover, due to the presence of noise, no exact solution is possible in general. The usual approach taken under such situations is to choose a vector x which optimizes some suitable performance criterion. Depending on the constraints put on the solution space, one obtains different solutions to the deconvolution problem in the presence of noise.

In data communications, the most sought after criterion is the minimization of the error probability. For the case of equally likely data vector, it can be shown that the maximum likelihood (ML) sequence estimation provides minimum probability of error [7]. The ML estimate of x is that vector which maximizes the conditional density $p(i/x)$.

Under the assumptions made in section 2.1 the conditional density function of r given a particular data vector x can be shown to be

$$p(r/x) = \prod_{i=1}^n \frac{1}{\sqrt{\pi N_0}} \exp\left(-\frac{1}{N_0} \left(r_i - \sum_{k=0}^g h_k x_{i-k}\right)^2\right)$$

It is easy to see that the above density function is maximized by minimizing the quadratic function

$$\begin{aligned} F(x) &= (\underline{r} - H\underline{x})^T (\underline{r} - H\underline{x}) \\ &= \sum_{i=1}^m \sum_{j=1}^n x_i x_j a_{|i-j|} - \sum_{i=1}^m b_i x_i + \sum_{i=1}^m x_i^2 \end{aligned} \quad (2.8)$$

where $a_{|i-j|}$ is the (i, j) th element of $H^T H$ and b_i is the i th element of the vector $2H^T \underline{r}$.

Therefore the deconvolution problem in our case is equivalent to minimization of the quadratic function given by (2.8) subject to the condition that the unknown variables take values from a finite alphabet. Since H is a convolution matrix it is easy to see that $H^T H$ is a symmetric Toeplitz matrix. We wish to note that a square matrix D is called Toeplitz if $d_{ij} = d_{|i-j|}$ where d_{ij} is its (i, j) th element. In addition as g is less than n , $H^T H$ is also a banded matrix. Thus the deconvolution problem is essentially that of minimizing a banded Toeplitz quadratic function subject to different constraints on the unknown variables depending on various performance criteria.

CHAPTER 3

MAXIMUM LIKELIHOOD DECONVOLUTION

In this chapter we use Mitten's lemma [55] to derive a maximum likelihood recursive deconvolution algorithm (ML-RDA) which essentially performs the job of Viterbi algorithm (VA) [38]. Mitten's lemma may be viewed as a more formal statement of Bellman's principle of optimality [56] and it clearly stipulates the conditions for optimal decomposition of multistage optimization problems. For the sake of easy reference Mitten's lemma is given in Appendix A.

The chapter is organized in the following manner. In Section 3.1 we use Mitten's lemma to develop a general recursive algorithm for minimizing decomposable nonlinear function of several variables. A specific case when the nonlinear function is a quadratic function is taken up in Section 3.1.1. It is shown that minimization of a quadratic function of real variables using the recursive algorithm is equivalent to Gaussian elimination algorithm for solving a set of linear simultaneous equations. The discrete version of the algorithm is discussed in Section 3.1.2. The ML-RDA for a banded Toeplitz quadratic function (BTQF) is given in Section 3.2. For the purpose of comparison in Section 3.3 we discuss VA.

for minimizing BTDF Strategies for alleviating storage and computational requirements are discussed in Section 3.4 Application of these strategies to Binary case is considered in detail in Section 3.4.1 Application of the recursive algorithm for decoding of block error correcting code is considered in Section 3.5

3.1 RECURSIVE ALGORITHM FOR MINIMIZATION OF NONLINEAR FUNCTION OF SEVERAL VARIABLES

In order to illustrate the principle of ML RDA we first give a method for minimizing a nonlinear function of n variables Let $F(x_1, x_2, \dots, x_n)$ be a function to be minimized We assume that the function is decomposable in the manner indicated below

$$F_1(\underline{x}) = F(x_1, x_2, \dots, x_n) = U_1(x_1, x_2, \dots, x_n) + V_1(x_2, x_3, \dots, x_n) \quad (3.1)$$

Note that V_1 is a function of x_2, x_3, \dots, x_n alone Now suppose that instead of minimizing $F(\underline{x})$ with respect to \underline{x} we can minimize U_1 with respect to x_1 first and then minimize the combined function $\min_{x_1} U_1(x_1, x_2, \dots, x_n) + V_1(x_2, x_3, \dots, x_n)$ with respect to x_2, x_3, \dots, x_n Put differently we wish to compute P_2 where

$$P_2 = \min_{x_2, \dots, x_n} [V_1(x_2, x_3, \dots, x_n) + \min_{x_1} U_1(x_1, x_2, \dots, x_n)] \quad (3.2)$$

At this stage a valid question to ask is whether or not $P_2 = \min_{x_1, \dots, x_n} F_1(x_1, x_2, \dots, x_n)$? If the answer is yes then effectively we have decomposed the n variable minimization problem into a single variable minimization problem followed by an $n-1$ variable minimization problem. Thus it is essential to know whether we can distribute the \min operator as indicated in (3.2) without affecting the optimum solution.

In Lemma 3.1 we show that whenever $F_1(x_1, \dots, x_n)$ is decomposable in the manner given by (3.1)

$$P_2 = \min_{x_1, \dots, x_n} F_1(x_1, x_2, \dots, x_n)$$

which effectively proves that the \min operator can be distributed in the manner shown in (3.2). We wish to note that Lemma 3.1 is a special case of the more general decomposition result given by Mitten [55] for multivariable optimization problems. A proof of Mitten's result for two variables may be found in [56].

Lemma 3.1 Given (3.1) and (3.2) $P_2 = \min_{x_1, \dots, x_n} F_1(x_1, x_2, \dots, x_n)$

Proof Let $P_1 = \min_{x_1, \dots, x_n} F_1(x_1, x_2, \dots, x_n)$

From the definition of minimum value we have

$$P_2 \geq P_1 \tag{3.3}$$

Whenever $U_1(x_1, x_2, \dots, x_n) < U_1(x_1, x_2, \dots, x_n)$ with x_2, x_3, \dots, x_n fixed it follows from (3.1) (since $V_1(x_2, x_3, \dots, x_n)$ is fixed) that

$$F_1(x_1, x_2, \dots, x_n) < F_1(x_1, x_2, \dots, x_n) \quad (3.4)$$

But for each value of x_2, x_3, \dots, x_n

$$U_1(x_1, x_2, \dots, x_n) = \min_{x_1} U_1(x_1, x_2, \dots, x_n) \leq U_1(x_1, x_2, \dots, x_n)$$

where x_1 is the value of x_1 at which $U_1(x_1, x_2, \dots, x_n)$ attains minimum value when x_2, x_3, \dots, x_n are fixed

Therefore from (3.4) we get

$$F_1(x_1, x_2, \dots, x_n) \leq F_1(x_1, x_2, \dots, x_n)$$

for all x_1 with x_2, x_3, \dots, x_n fixed (since $V_1(x_2, x_3, \dots, x_n)$ is fixed)

Hence

$$\begin{aligned} F_1(x_1, x_2, x_3, \dots, x_n) &\leq \min_{x_1} F_1(x_1, x_2, \dots, x_n) \\ \text{Hence } P_2 &= \min_{x_2, \dots, x_n} [V_1(x_2, x_3, \dots, x_n) + \min_{x_1} U_1(x_1, x_2, \dots, x_n)] \\ &= \min_{x_2, \dots, x_n} F_1(x_1, x_2, \dots, x_n) \leq \\ &\quad \min_{x_2, \dots, x_n} \min_{x_1} F_1(x_1, x_2, \dots, x_n) = P_1 \end{aligned}$$

therefore $P_2 \leq P_1$

From (3) we thus get $P_2 = P_1$

**

Let x_1 be the value of x_1 which minimizes $U_1(x_1, x_2, \dots, x_n)$ for a fixed x_2, \dots, x_n . Note that x_1 becomes a function of x_2, x_3, \dots, x_n . Let

$$x_1 = g_1(x_2, x_3, \dots, x_n)$$

Substituting for x_1 in $F_1(x_1, x_2, \dots, x_n)$ the minimization problem now reduces to a $n-1$ variable minimization problem

$$\min_{x_2, \dots, x_n} F_2(x_2, x_3, \dots, x_n) \quad \text{where} \quad F_2(x_2, x_3, \dots, x_n) = F_1(g_1(x_2, x_3, \dots, x_n), x_2, \dots, x_n)$$

Assuming again that F_2 is decomposable we can write F_2 as

$$F_2(x_2, x_3, \dots, x_n) = U_2(x_2, x_3, \dots, x_n) + V_2(x_3, x_4, \dots, x_n)$$

Using Lemma 3.1 we can write

$$\min_{x_2, \dots, x_n} F_2(x_2, x_3, \dots, x_n) = \min_{x_3, \dots, x_n} [V_2(x_3, x_4, \dots, x_n) + \min_{x_2} U_2(x_2, x_3, \dots, x_n)]$$

Let x_2^* be the value of x_2 which minimizes $U_2(x_2, x_3, \dots, x_n)$. Then

$$x_2^* = g(x_3, x_4, \dots, x_n)$$

The recursive procedure now can be easily written for the i th stage as

$$F_1(x_1, x_{1+1}, \dots, x_n) = F_{1-1}(x_1, x_2, \dots, x_{1-1}^*, x_1, \dots, x_n)$$

$$\min_{x_1, \dots, x_n} F_1 = \min_{x_{1+1}, \dots, x_n} [V_1(x_{1+1}, \dots, x_n) + \min_{x_1} U_1(x_1, \dots, x_n)]$$

$$x_1 = g_1(x_{1+1}, \dots, x_n)$$

$$i = 1, 2, \dots, n$$

At the n th stage we obtain

$$\min_{x_n} F_n(x_n) = \min_{x_n} U_n(x_n)$$

Therefore x_n^* is a constant i.e. g_n is a constant function. Substituting x_n^* in $g_{n-1}(x_{n-1}, \dots, x_n)$ x_{n-1} can be obtained. Continuing this procedure of back substitution all the x_i $i = 1, 2, \dots, n$ are obtained.

It is not difficult to see that the foregoing approach is very similar in spirit to the Gaussian elimination method of solving linear simultaneous equations which involves forward elimination and back substitution. The collection of functions

$$x_1^* = g_1(x_2, x_3, \dots, x_n)$$

$$x_2^* = g_2(x_3, x_4, \dots, x_n)$$

$$x_1^* = g_1(x_{1+1}, x_{1+2}, \dots, x_n)$$

$$x_n = g_n \text{ constant}$$

is reminiscent of the upper triangular matrix which results after forward elimination. In view of the foregoing observation we wish to call the recursive algorithm developed above a generalized Gaussian elimination procedure.

In the preceding algorithm we implicitly assumed that the indicated single variable minimization is always possible. That is, $\min_y U_1(x_1, \dots, x_n)$ exists for all i and fixed x_{i+1}, \dots, x_n . However, this may not always be possible for an arbitrarily specified nonlinear function. In addition, single variable minimization itself may be difficult to carry out. Nevertheless, this approach has several advantages in the case of the ML-deconvolution problem formulated in Sec. 2.4. Before discussing that case we wish to consider the problem of minimizing a quadratic function of n real variables as an illustration of the foregoing recursive algorithm.

3.1.1 Real Variable Case

For the sake of simplicity we consider $n = 3$. Generalization to n variables is straightforward. Let

$$F_1(x_1, x_2, x_3) = -2 \sum_{i=1}^3 x_i b_i + \sum_{i=1}^3 \sum_{j=1}^3 x_i x_j a_{ij} + \sum_{i=1}^3 r_i^2$$

$$\text{and } a_{ij} = a_{ji} \text{ for all } i, j$$

where b_i , a_{ij} and r_i , $i = 1, 2, 3$ are constants. Carrying out the decomposition of F_1 with respect to x_1 we obtain

$$F_1 = U_1(x_1, x_2, x_3) + V_1(x_2, x_3)$$

$$\text{where } U_1(x_1, x_2, x_3) = -2x_1b_1 + 2x_1 \sum_{j=2}^3 x_j a_{1j} + x_1^2 a_{11}$$

x_1^* which minimizes U_1 with x_2, x_3 fixed is given by

$$x_1 = (b_1 - \sum_{j=2}^3 x_j a_{1j}) / a_{11}$$

we assume that $a_{11} \neq 0$. Substituting x_1^* in F_1 we obtain

$$F_2(x_2, x_3) = -2 \sum_{i=2}^3 x_i b_i + \sum_{i=2}^3 \sum_{j=2}^3 x_i x_j a_{ij} + \sum_{i=2}^3 r_i^2 + (r_1)^2$$

$$\text{where } b_i = b_i - b_1 a_{i1} / a_{11} \quad i = 2, 3$$

$$(r_1)^2 = r_1^2 - b_1^2 / a_{11}$$

$$\text{and } a_{ij} = a_{ij} - a_{i1} a_{j1} / a_{11} \quad i, j = 2, 3$$

$$\text{now } F_2(x_2, x_3) = U_2(x_2, x_3) + V_2(x_3)$$

$$\text{where } U_2(x_2, x_3) = -2x_2b_2 + 2x_2x_3a_{23} + x_2^2a_{22}$$

$$\text{therefore } x_2^* = (b_2 - x_3a_{23}) / a_{22} \quad a_{22} \neq 0$$

$$\text{using } x_2^* \text{ we obtain } F_3(x_3) = -2x_3b_3 + x_3^2a_{33} + (r_1)^2 + (r_2)^2 + r_3^2$$

minimization of F_3 gives

$$x_3 = b_3 / a_{33} \quad a_{33} \neq 0$$

Back substitution of x_3^* gives x_2^* which in turn gives x_1^*

The foregoing example clearly shows that the recursive algorithm for minimization of a quadratic function of n variables is equivalent to Gaussian elimination technique for solving a set of linear equations

3.1.2 Discrete Variable Case

Having discussed certain aspects of the multivariable minimization with real variables we now consider the case when the variables take values from a finite set. Without any loss of generality we consider the quadratic function used in the previous section with the difference that the variables take values from the set $S = \{-3, -2, -1, 0, 1, 2, 3\}$. Extension to n -variable case and any given finite alphabet is straightforward.

Following the steps as in the real variable case in the first stage we have to find $\min_{x_1} U_1(x_1, x_2, x_3)$. In this case differentiation is not possible. Hence we must tabulate values of x_1 which minimize U_1 for fixed x_2 and x_3 . The tabulation format is shown in Table 3.1. No. of entries in the first column of the table will be 7^2 corresponding to 7^2 possible values of x_2 and x_3 . Moreover for each set of x_2 and x_3 we must check 7 values of x_1 to decide x_1^t . Thus the first stage itself involves 7^3 evaluations of U_1 . Further in general it is not possible to express x_1^t in a closed form as in the case of real variables. Therefore x_1 can not be substituted in F_1 to obtain F_2 . This implies that we must use the table for x_1^t in the second stage. In the second stage we write F_2 as

$$F_2(x_2, x_3) = J_1^t(x_1, x_2, x_3) + U_2(x_2, x_3) + V_2(x_3)$$

Tabl 3 1 Stage 1

x_2	x_3	x_1^*	U_1
3	-3	-	-
3	-2	-	-
3	-1	-	-
0	3	-	-

Tabl 3 2 Stage 2

x_3	x_2	J_2
-3	-	-
-2	-	-
-1	-	-
0	-	-
1	-	-
2	-	-
3	-	-

For each value of x_2 and x_3 we can use U_1 from the table for x_1^* . Thus in the second stage we minimize $U_1 + U_2(x_2, x_3)$ with respect to x_2 by fixing x_3 . There are 7 different values for x_3 and for each x_3 x_2^* is to be decided on the basis of 7 possible values of x_2 . Thus the second table will have 7 entries as shown in Table 3 2. In a similar fashion we can carry out the computation required in stage 3 with the help of Table 3 2 of stage 2.

Let us now examine the computational complexity for n -variable case with each variable taking q discrete values. In the first stage q^n evaluations of U_1 is required. Likewise in the i th stage q^{n-i+1} evaluation of $U_1 + U_{1-i}^*$ is required. On the other hand a direct search method would require q^n evaluation of

th function F . It thus becomes clear that for the scheme described to become computationally efficient the functional forms of U_1, U_2, \dots, U_n must be much simpler than the functional form of F . In other words U_1, U_2, \dots, U_n must have the following desirable properties

- 1 U_i 's must be functions of a small number of variables compared to the total number of variables involved in the minimization problem
- 2 U_i 's must be easily computable. By this we mean the number of addition and multiplications needed for evaluating the function must be small

When both the properties are satisfied by an objective function it is possible to reduce the number of computations and storage requirements substantially using the foregoing recursive algorithm compared to the direct enumeration approach. It may however be noted that storage requirement is still large compared to the enumeration approach.

The objective function in the case of ML-deconvolution problem formulated in Section 2.4 is a special type of quadratic function called banded Toeplitz quadratic function. Such a quadratic function meets the above mentioned requirements. In the next section we derive a ML-RDA for this case.

3.2 RECURSIVE ALGORITHM FOR BANDED TOEPLITZ QUADRATIC FUNCTIONS

We recall from Section 2.4 that the function

$$F(x_1, x_2, \dots, x_m) = 2 \sum_{i=1}^m x_i b_i + \sum_{i=1}^m \sum_{j=1}^m x_i x_j a_{|i-j|} + c$$

where $a_{|i-j|} = 0$ for $|i-j| > \alpha$, $0 \leq \alpha < m$

is called banded Toeplitz quadratic function (BTQF) for the matrix formed by $a_{|i-j|}$ as elements is a banded Toeplitz matrix.

Let $F_1(\underline{x}) = F(x)$. For the sake of uniformity, F_1 can be decomposed as

$$\begin{aligned} F_1 &= -2x_1 b_1 + 2x_1 \sum_{i=2}^{g+1} x_i a_{|i-1|} + x_1^2 a_0 \\ &\quad + \left[-2 \sum_{i=2}^m x_i b_i + \sum_{i=2}^m \sum_{j=2}^m x_i x_j a_{|i-j|} \right] \\ &= U_1(x_1, x_2, \dots, x_{g+1}) + V_1(x_2, x_3, \dots, x_m) \end{aligned}$$

Note that U_1 is only a function of $g+1$ variables. Further, U_1 is a function of consecutive $g+1$ variables rather than $g+1$ arbitrary variables as it may happen in the case of a general objective function. This is a special property of BTQF. In addition, the form of U_1 at every stage is the same. Thus the same type of computation procedure is used in every stage. Through the following example we have tried to illustrate various aspects of the algorithm. The example is from [11].

Example 3.1

$$\text{Let } F_1(x_1, x_2, \dots, x_7) = -2 \sum_{i=1}^7 x_i b_i + \sum_{i=1}^7 \sum_{j=1}^7 x_i x_j a_{|i-j|}$$

$$\text{where } b_1 = 1.5, \quad b_5 = 1.5, \quad a_0 = 1, \quad a_1 = 0.5, \quad a_2 = -0.25$$

$$b_2 = 2.0, \quad b_6 = -3.0, \quad g = 2$$

$$b_3 = 0.5, \quad b_7 = 0.5$$

$$b_4 = 1.0$$

and $x_i, i = 1, 2, \dots, 7$ take +1 or -1 values

$$\text{Stage 1 } F_1(x_1, x_2, \dots, x_7) = U_1(x_1, x_2, x_3) + V_1(x_2, x_3, \dots, x_7)$$

$$\begin{aligned} U_1(x_1, x_2, x_3) &= -2x_1 b_1 + 2x_1 \sum_{i=2}^3 x_i a_{|1-i|} + x_1^2 a_0 \\ &= -3x_1 + 2x_1[0.5 - 0.25x_3] + x_1^2 \end{aligned}$$

Table 3.3 is obtained by determining the value of x_1 that minimizes U_1 for a fixed x_2 and x_3 .

Table 3.3 Stage 1

x_2	x_3	x_1^*	U_1^*
-1	1	1	-5/2
-1	1	1	-3.5
1	1	1	-0.5
1	1	1	-1.5

Since $x_1 = 1$ for all the values of x_2, x_3 we can decide at this stage itself that the first variable is $x_1^* = 1$

$$\text{Stage 2} \quad U_2(x_2, x_3, x_4) = -2x_2b_2 + 2x_2 \sum_{i=3}^4 x_i a_{i2} + x_2^2 a_{00} + U_1(x_2, x_3)$$

As the calculations are identical in all the stages we have presented the final results in Tables 3.4-3.8 corresponding to stages 2 to 6

Table 3.4 Stage 2

x_3	x_4	x_2^*	U_2
-1	-1	1	-4
1	1	1	-5
1	-1	1	-3
1	1	1	-4

Therefore the second variable
 $x_2^* = 1$

Table 3.5 Stage 3

x_4	x_5	x_3	U_3
-1	-1	1	-3.5
-1	1	1	-4.5
1	-1	1	-4.5
1	1	-1	-5.5

Table 3.6 Stage 4

x_5	x_6	x_4^*	U_4
-1	-1	1	-6
-1	1	1	-7
1	-1	1	-3
1	1	1	-4

Table 3.7 Stage 5

x_6	x_7	x_5^*	U_5
-1	-1	1	-7
-1	1	-1	-6.5
1	-1	-1	10.5
1	1	1	9.5

Table 3.8 Stage 6

x_6	x_7	U_6
1	-1	9.5
1	1	-12.5
1	-1	-2.5
1	1	-1.5

From Table 3.8 we see that $x_6 = 1$ and $x_7 = 1$. On back substitution Table 3.7 gives $x_5 = 1$. It may also be noticed that Table 3.6 and Table 3.7 for stage 4 and 5 respectively directly indicate that $x_4^* = 1$ and $x^t = -1$. Using the values of x_4^* and x^t in Table 3 we obtain $x_3 = -1$.

3.3 VITERBI ALGORITHM FOR MINIMIZING QF

For the purpose of comparison we have given a brief description of VA in this section.

Recall that the LQF minimization problem arose in connection with transmission of digital data through a finite impulse response channel modelled as finite energy transversal filter.

At the k th instant the output of such filter can be represented in terms of $x_k, x_{k-1}, \dots, x_{k-g}$ as

$$y_k = \sum_{i=0}^g h_i x_{k-i} \quad k = 1, 2, 3$$

where g is the memory of the transversal filter. Thus y_k is a function of the present input as well as the previous state of the transversal filter. Let S_k denote the state of the transversal filter at the k th instant. Then

$$S_k = (x_k, x_{k-1}, \dots, x_{k-g+1}) \quad x_k = 0 \text{ if } k < 0$$

In terms of S_k 's y_k can be expressed as

$$y_k = f(h_0, h_1, \dots, h_g, S_k, S_{k-1})$$

Therefore given an input sequence $\{x_k\}$ we have a corresponding state sequence $\{S_k\}$ and an output sequence $\{y_k\}$. This correspondence is one to one. This observation is crucial to the development of VA. Due to this correspondence we can minimize $F(\underline{x})$ over S_k 's instead of x_k 's. For this purpose decompose $F(\underline{x})$ as

$$F(\underline{x}) = f_1(\underline{x}) = \left[-2 \sum_{i=2}^n x_i b_i + \sum_{i=2}^m \sum_{j=2}^n x_i x_j a_{|i-j|} \right] +$$

$$\left[-2x_1 b_1 + 2x_1 (x_2 a_1 + x_3 a_2 + \dots + x_{g+1} a_g) + x_1^2 a_0 \right]$$

Note that the quantity inside the second pair of brackets is only a function of x_1, x_2, \dots, x_{g+1} . Let

$$f_1(\underline{x}) = u_1(x_1, \dots, x_{g+1}) + v_1(x_2, \dots, x_m)$$

It is easy to see that $u_1(x_1 \ x_2 \ \dots \ x_{g+1}) = u_1(S_{g+1} \ S_g)$

Now v_1 can be similarly decomposed as

$$v_1(x_2 \ x_3 \ \dots \ x_m) = v_2(x_3 \ x_4 \ \dots \ x_m) + u_2(S_{g+2} \ S_{g+1})$$

In a similar fashion we can write for the k th stage

$$v_k(x_{k+1} \ \dots \ x_m) = v_{k+1}(x_{k+1} \ \dots \ x_m) + u_k(S_{g+k} \ S_{g+k-1})$$

$$k = 1 \ 2 \ 3 \ \dots \ m-g$$

Therefore $\Gamma(\underline{x})$ can be written as

$$F(S_1 \ S_2 \ \dots \ S_n) = \sum_{k=1}^{m-g} u_k(S_{g+k} \ S_{g+k-1}) + v_{m-g}(S_m)$$

where

$$u_k(S_{g+k} \ S_{g+k-1}) = -2x_k b_k + 2x_k \sum_{j=k+1}^{g+k} x_j a_{j-k} + x_k^2 a_0$$

$$k = 1 \ 2 \ 3 \ \dots \ m-g$$

We wish to note that the derivation of the foregoing recursive equations is based on the approach given by Hayes [41]

The optimality of the algorithm follows from Lemma 3.1

The algorithm works as follows. In the first stage of the algorithm $u_1(S_{g+1} \ S_g)$ is minimized with respect to S_g for fixed S_{g+1} . As S_{g+1} can assume any one of the q^g states we have to store u_1^* value corresponding to q^g states. As S_g and S_{g+1} differ in only one place, determination of optimum S_g given S_{g+1} is effectively one variable optimization. To complete our discussion let

$$F(S_g, S_m) = f_1(S_g, S_1) = u_1(S_{g+1}, S_g) + \sum_{k=2}^{n-g} u_k(S_{g+k}, S_{g+k-1}) + v_{m-g}(S_m)$$

Hence

$$\min_{S_g} f_1 = \min_{S_{g+1}} \left[\sum_{k=2}^{n-g} u_k(S_{g+k}, S_{g+k-1}) + v_{m-g}(S_m) + \min_{S_g} u_1(S_{g+1}, S_g) \right]$$

Define $u_k^* = u_k(S_{g+1}, S_{g+k-1})$ $u_0^* = 0$

Then the recursive equations for f_1 can be written as

$$u_k^* = \min_{S_{g+k-1}} [u_k(S_{g+1}, S_{g+k-1}) + u_{k-1}^*] \quad k = 1, 2, \dots, m-g$$

Finally the optimum value of F is given as

$$F^* = \min_{S_m} [u_{m-g}^* + v_{m-g}(S_m)]$$

Result obtained by applying VA to Example 3.1 is shown in the Figure 3.1. In the figure we have shown the survivor paths through the trellis. In this figure dotted lines indicate alternate survivors giving same function values. From the figure it may be noted that (1,1) is a node common to all the survived paths in stage 3. This implies that $\lambda_1 = 1$. Thus decisions regarding data can be made even before completing the forward recursion process. This is due to the merging of the survivor paths through the Trellis.

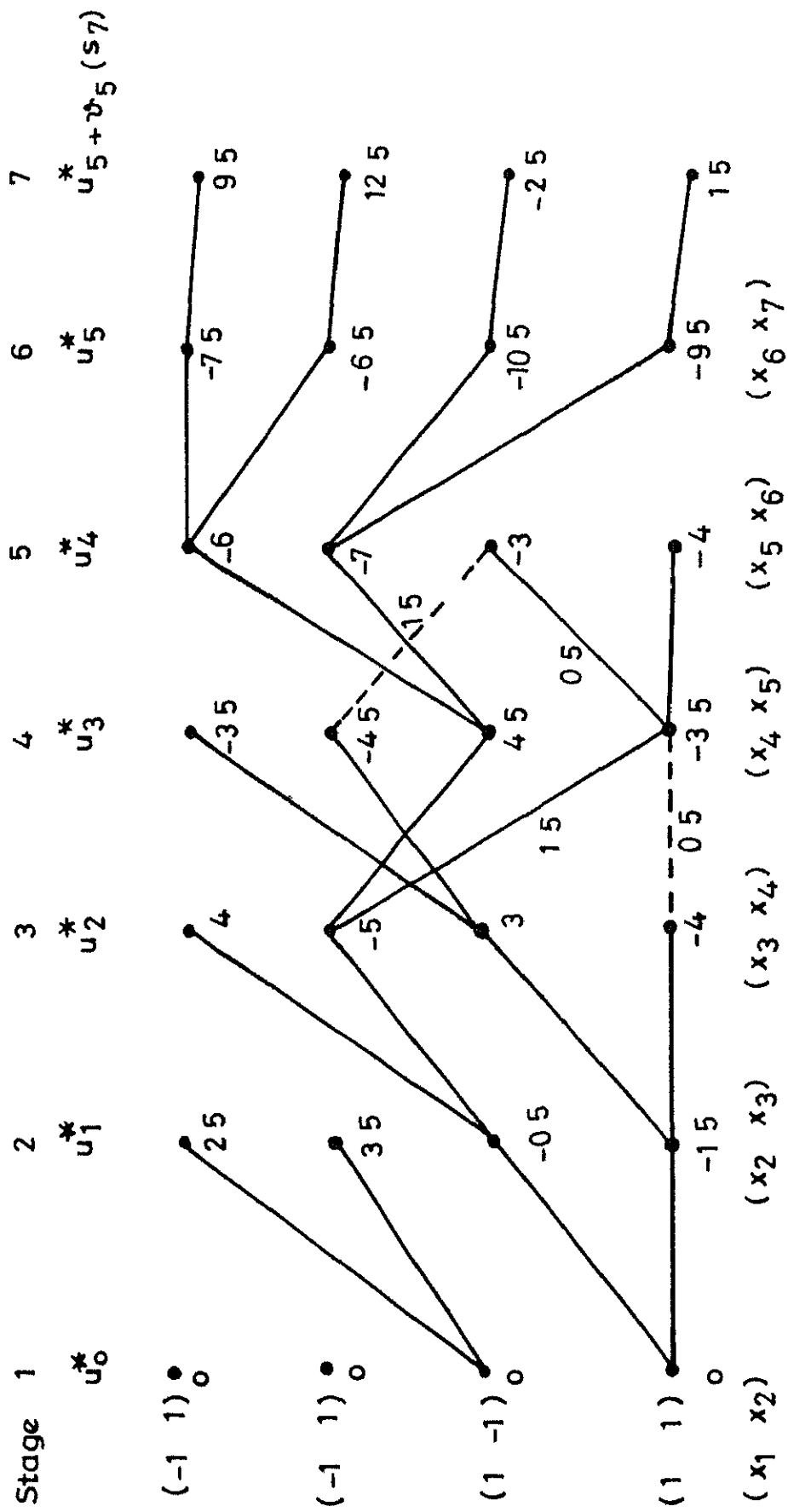


FIG 3 1 Trellis diagram showing survivor paths

3 4 TRATECIES FOR REDUCING COMPUTATIO AL AND STORAGE
REQUIREMENTS

As the procedure is same for a general n-variable BTQF we see that the storage requirement is q^g registers of length $(m+1)$ each for storing x_i^* s at each stage and the accumulated minimum

Stage 1

x_1	x_2	x_3	U_1^t
1	-1	-1	-5/2
1	-1	1	-3 5
1	1	-1	-0 5
1	1	1	-1 5

Stage 2

x_1^*	x_2	x_3	x_4	U_2
1	1	-1	-1	-4
1	1	-1	1	-5
1	1	1	-1	-3
1	1	1	1	-4

Stage 3

x_1^*	x_2^*	x_3^*	x_4	x_5	U_3^*
1	1	1	1	-1	-3 5
1	1	1	-1	1	-4 5
1	1	1	1	-1	-4 5
1	1	-1	1	1	-3 5

Stage 6

x_1^*	x_2	x_3^*	x_4^t	x_5^t	x_6	x_7	U_6^t	State	
1	1	1	1	-1	-1	-1	-9 5	-1	-1
1	1	1	1	-1	-1	1	-12 5	-1	1
1	1	-1	1	-1	1	-1	-2 5	1	-1
1	1	-1	1	-1	1	1	-1 5	1	1

Figure 3 2 Developm nt of the tables in ML-RDA for Example 3 1

value at every stage. Thus the storage requirement grows linearly with the block length. Further, since at every stage we need q^g function evaluations, the computational complexity of the algorithm grows linearly with block length and exponentially with channel memory.

Even with the foregoing strategy for storing the tables, the algorithm still requires a large amount of storage and computations for even moderate block lengths. Thus, if the recursive algorithm is to become acceptable, it must have additional features for minimising storage and computational requirements. Towards this end, we now propose and examine a method for reducing the complexity of ML-RDA, which is expected to result in a greater savings in storage and computations.

To begin with, let us examine Tables 3.3-3.8 for Example 3.1. For instance, consider Table 3.5 obtained at the 3rd stage. Note that if we are able to express x_3^* as a function of x_4 and x_5 , substituting for x_3^* in the objective function eliminates the need for the table in the next stage computations. Moreover, for the purpose of back substitution, we need retain only the functional form of x_3 . In this case, it is easy to see that

$$x_3^* = -x_4$$

If at every stage we are able to obtain such simple representation, storage requirements are reduced to a minimum. In the example under consideration, all the functions have simple representations as given below.

tag	1	2	3	4	5	6	
Functions	$x_1=1$	$x_2=1$	$x_3^*=-x_4$	$x_4=1$	$x_5^*=-1$	$x_6=1$	and $x_7=1$

such simple functional forms cannot always be guaranteed. For the time being assume that such representations are possible. Then the next question would be can we arrive at the functional form without enumerating all possible values of the fixed variables? If this is possible we would also have achieved reduction in the computational complexity.

However in general this appears to be a difficult problem. In the first place functional form of x_i may be complicated due to which storage of such function as well as substitution into the objective function may be difficult. Secondly it may be difficult to directly obtain the functional forms. Therefore to begin with instead of considering the most general case we study the binary case in the next section.

4.1 Binary Data Transmission

When the data symbols assume one of the two values 1 or -1 the objective function can be converted into another function called pseudo-Boolean function (PBF) [57] which is a function of Boolean variables with real coefficients. For this we use the transformation $y = (1+x)/2$ (which means y takes 0 and 1 as x takes -1 and 1 respectively). With the help of this transformation the quadratic function $F(x)$ can be converted into a PBF $F(y)$ where y_1, y_2, \dots, y_m are Boolean variables.

Let $F(\underline{x})$ expressed in the matrix vector form as

$$F(\underline{x}) = -2\underline{x}^T \underline{b} + \underline{x}^T \underline{A} \underline{x}$$

We use the transformation $\underline{y} = (\underline{1} + \underline{x})/2$

where $\underline{1}$ is a vector of all ones. With this transformation we obtain

$$\Gamma(\underline{y}) = -2\underline{y}^T \underline{b} + \underline{y}^T \underline{A} \underline{y} + p$$

where $\underline{b} = 2(\underline{b} + \underline{A}\underline{1})$ $\underline{A} = 4\underline{A}$ $p = 2\underline{1}^T \underline{b} + \underline{1}^T \underline{A} \underline{1}$

The constant term p can be ignored for the purpose of minimizing $F(\underline{y})$

$\Gamma(\underline{y})$ is same as $\Gamma(\underline{x})$. Therefore the IL-PDA given in Section 3.2 applies to this case with the difference that the tables at each stage now contains Boolean variables. As our interest is in expressing x_1^* in terms of x_{1+1} x_{1+2} x_{1+g} at the i th stage we observe that the table in this case is a truth table for x_1^* in terms of x_{1+1} x_{1+2} x_{1+g} . Now using results from the theory of Boolean functions [72] we can represent x_1 in some convenient form. For instance minimal sum of product or product of sums form.

Thus we have shown that in the case of binary data there exists a procedure for expressing x_1^* in terms of x_{1+1} x_{1+2} x_{1+g} in the i th stage. However simple expressions for x_1^* may not always be obtained. Let us consider the worst case where x_1^*

is represented in terms of $x_{1+1} \dots x_{1+g}$. The truth table for x_1^* will contain 2^g entries. In this case the storage can be reduced to some extent by obtaining a minimal representation of x_1^* . Methods such as Quine-McCluskey or any such similar method [72] could be used for this purpose. However we may have to use a special purpose computer which implement Boolean function minimization algorithm.

Now we consider the problem of obtaining functional representation for y_1^* without enumerating all possible 2^d cases. By using an algorithm developed by Hammer et al. [57] we show that the computational complexity can be reduced at every stage without sacrificing optimality. Towards this end we use Lemma 3.1 to derive a recursive algorithm developed originally by Hammer et al. [57] for minimizing nonlinear PBF's in the next section. The approach taken here is much simpler compared to the derivation given in [57].

3.4.2 Minimization of Pseudo-Boolean Functions

Consider a PBF $F_1(x_1, x_2, \dots, x_m)$ of m variables. Since pseudo-Boolean function is linear in each of its variables we can write

$$F_1(x_1, x_2, \dots, x_m) = x_1 U_1(x_2, x_3, \dots, x_m) + V_1(x_2, x_3, \dots, x_m)$$

Application of Lemma 3.1 gives

$$\min_{x_1} \min_{x_m} F_1 = \min_{x_1} \min_{x_m} [V_1(x_2, x_3, \dots, x_m) + \min_{y_1} x_1 U_1(x_2, x_3, \dots, x_m)]$$

Now since x_1 's are Boolean variables it follows that

$$x_1 = \begin{cases} 0 & \text{if } U_1(x_2, x_3, \dots, x_m) > 0 \\ \emptyset & \text{if } U_1(x_2, x_3, \dots, x_m) = 0 \\ 1 & \text{if } U_1(x_2, x_3, \dots, x_m) < 0 \end{cases}$$

where \emptyset is a don't care variable (x_1 arbitrary)

Therefore a Boolean function representation for x_1 can be obtained by determining those x_2, x_3, \dots, x_m for which $U_1 < 0$. Substitution of x_1^* into F_1 gives F_2 . The other stages of the minimization follow the same pattern as in the case of recursive procedure described earlier. Thus at the i th stage we have

$$\min_{x_1} \min_{x_m} F_1 = \min_{x_{i+1}} \min_{x_m} [V_i(x_{i+1}, x_{i+2}, \dots, x_m) + \min_{y_i} x_i U_i(x_{i+1}, x_{i+2}, \dots, x_m)]$$

and

$$y_i^* = \begin{cases} 0 & \text{if } U_i > 0 \\ \emptyset & \text{if } U_i = 0 \\ 1 & \text{if } U_i < 0 \end{cases}$$

As we are interested in only one solution \emptyset can be fixed as 0 or 1 and we can solve $U_1 \leq 0$ or obtaining x_1 or $U_1 \geq 0$ or obtaining x_1^* . Hammer et al [57] have also developed an algorithm for determining all solutions which satisfy $U_1 \leq 0$. This algorithm can be used to reduce the computational complexity at each stage.

From the foregoing discussion it is clear that in the binary case it is possible to reduce storage and computational complexity of the RDA if fast processors are available for minimization of Boolean functions and for implementing Hammer et al procedure for obtaining x_1 .

It may be noted that determination of minimal representation of Boolean functions is a classical problem in the theory of switching functions. Although a large number of methods are available [72] no definite claim can be made in favour of one approach over the other.

In summary the overall efficiency of the RDA as implemented in the manner described in this section depends to a great extent on the availability of fast algorithms for obtaining minimal representations for x_1^* either in the Boolean function form or in the pseudo-Boolean function form.

In the nonbinary case however no such claims can be made. This aspect needs further investigation.

3.5 RECURSIVE ALGORITHM FOR DECODING BLOCK ERROR CORRECTING CODES

As an application of the recursive algorithm developed in the previous section we consider maximum likelihood decoding of binary block codes. Our approach consists in formulating the ML decoding problem as an equivalent problem of minimizing a pseudo-Boolean function of several variables. It may be noted that the algorithm developed in this section can also be used for maximum likelihood decoding of convolution codes.

We wish to note that Volf [73] has studied the application of VJ to decode block codes by constructing Trellis diagrams. The approach considered here does not require Trellis construction and it seems to provide a more general method for maximum likelihood decoding of binary block codes.

The codes considered here are over $GF(2)$ where $GF(2)$ denotes Galois field with two elements 0 and 1 [72]. For such codes the ML decoding is equivalent to solving the set of linear equations $C\underline{x} = \underline{r}$ over $GF(2)$ such that the Hamming distance between $G\underline{x}$ and \underline{r} is minimum when all the code words are assumed to equiprobable. The matrix G is called the generator matrix of the code in the parlance of coding theory [6] [7]. The received vector \underline{r} is an additive combination of the transmitted codevector which is not observable at the receiver and the noise vector \underline{w} . That is $\underline{r} = \underline{c} \oplus \underline{w}$ where \oplus denotes addition

operation in $GF(2)$. It may be noted that the elements of \underline{x} and \underline{y} are either 0 or 1. We also assume that C is an $n \times m$ matrix.

The Hamming distance between two vectors \underline{x} and \underline{y} over $GF(2)$ is defined as the number of nonzero entries in $\underline{x} \oplus \underline{y}$. It is also known as the weight of $\underline{x} \oplus \underline{y}$ and noted as $w(\underline{x} \oplus \underline{y})$.

Since $GF(2)$ elements are also elements of the real field we can show that

$$\underline{p} \oplus \underline{q} = \underline{p} + \underline{q} - 2\underline{p} \underline{q}$$

where $\underline{p} \underline{q}$ denote pointwise multiplication. We now make use of the property that if \underline{y} is 0-1 real vector then

$$w(\underline{y}) = \underline{1}^T \underline{y}$$

where $\underline{1}$ is column vector of all ones. Therefore it is easy to see that

$$w(\underline{p} \oplus \underline{q}) = w(\underline{p} + \underline{q} - 2\underline{p} \underline{q}) = w(\underline{p}) + w(\underline{q}) - 2w(\underline{p} \underline{q})$$

Thus maximum likelihood decoding problem can be formulated as

$$\begin{aligned} & \min_x (w(G\underline{x}) + w(\underline{r}) - 2w(G\underline{x} \underline{r})) \\ & = \min_x (\underline{1}^T G\underline{x} + \underline{1}^T \underline{r} - 2\underline{1}^T (G\underline{x} \underline{r})) \end{aligned} \quad (3.5)$$

Note that Gx is to be computed over $GF(2)$. Therefore we need a suitable representation for the elements of Gx . Let a_1, a_2, \dots, a_n denote n rows of G . Then we can write Gx as $(a_1x, a_2x, \dots, a_nx)$ where $a_1x = a_{11}x_1 \oplus a_{12}x_2 \oplus \dots \oplus a_{1m}x_m$. We can use the relation

$$u \oplus v = u + v - 2uv \quad (3.6)$$

in order to convert a_1x into an equivalent pseudo-Boolean function. However such a representation becomes too complicated when more variables are involved. A more compact form of representation is necessary to obtain a representation which reflects the generator matrix explicitly. Towards that end consider the transformation $u = 1-2u$ and $v = 1-2v$. Substituting for u and v on the right hand side of (3.6) we obtain $u \oplus v = (1 - uv)/2$ which is a more compact expression compared to $u+v-2uv$. We can further show that

$$u \oplus v \oplus w = (1 - uvw)/2 \quad \text{where } w = 1-2w$$

In general we have $\sum_{i=1}^m c_i x_i = (1 - \prod_{i=1}^m z_i^{c_i})/2$ where \sum denotes summation in $GF(2)$ and $c_i \in GF(2)$, $i = 1, 2, \dots, m$ and $z_i = 1-2x_i$. Now for (x) we have

$$a_1x = \frac{1}{2} \left(1 - \prod_{j=1}^m z_j^{a_{1j}} \right) \quad i = 1, 2, \dots, m$$

$$z_j^{a_{1j}} = \begin{cases} 1 & \text{if } a_{1j} = 0 \\ z_j & \text{if } a_{1j} = 1 \end{cases}$$

Note that the representation for $\underline{a}_1 \underline{x}$ clearly shows the effect of a_{1j} s

Going back to the minimization problem given by

(3.5) // can ignore $\underline{1}^T \underline{x}$ for the purpose of minimization as it is a constant. The rest of the terms can be combined as

$$\min_{\underline{x}} (\underline{1}^T \underline{G} \underline{x} \underline{r}) \quad \text{where } \underline{r} = \underline{1} - 2\underline{x}$$

Let $\underline{d} = (d_1 \ d_2 \ \dots \ d_n)^T$ where $d_j = \sum_{i=1}^m z_i^{a_{ij}} \quad j = 1, 2, \dots, n$

$$\text{Then } \underline{G} \underline{x} \underline{r} = \frac{1}{2} (\underline{1} - \underline{d})^T \underline{x} = \frac{1}{2} \underline{1}^T \underline{x} - \frac{1}{2} \underline{d}^T \underline{x}$$

Thus we have to find

$$\min_{\underline{x}} \left(-\frac{1}{2} \underline{1}^T (\underline{d} \underline{x}) \right) = \min_{\underline{x}} \left(-\frac{1}{2} \sum_{j=1}^n d_j r_j \right) \quad \text{where } r_j = 1 - 2x_j$$

With the objective function in this form we can now proceed with the recursive optimization algorithm discussed in Section 3.4.1. To round up the foregoing discussion we give three illustrative examples

Example 3.2 (Hamming Code) Consider the generator matrix G given by

$$G^T = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Let $\underline{x} = (1 \ 1 \ 1 \ 0)^T$ be the transmitted vector Then

$\underline{c} = (1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0)^T$ is the transmitted code vector

Let $\underline{r} = (0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0)^T$ be the received vector For this situation we have

$$d_1 = z_1 \quad d_2 = z_2 \quad d_3 = z_3 \quad d_4 = z_4 \quad d_5 = z_1 z_2 z_4$$

$$d_6 = z_1 z_3 z_4 \quad \text{and} \quad d_7 = z_2 z_3 z_4$$

Therefore the function to be minimized is given by

$$f(\underline{z}) = -\frac{1}{2} (z_1 - z_2 - z_3 + z_4 - z_1 z_2 z_4 + z_1 z_3 z_4 + z_2 z_3 z_4)$$

Neglecting $1/2$ we can write $f(\underline{z})$ as

$$f(\underline{z}) = z_1(-1 - z_2 z_4 - z_3 z_4) - z_2 - z_3 + z_4 + z_2 z_3 z_4 =$$

$$U_1(z_1 \ z_2 \ z_3 \ z_4) + V_1(z_2 \ z_3 \ z_4)$$

Minimization of U_1 with respect to z_1 for fixed $z_2 \ z_3$ and z_4 gives the following table for stage 1

Stage 1

z_2	z_3	z_4	z_1^*	U_1^*
-1	-1	-1	1	-3
-1	-1	1	1	-1
-1	1	-1	1	-1
-1	1	1	1	-1
1	-1	-1	1	-1
1	1	1	1	-1
1	1	-1	-1	-1
1	1	1	1	-3

Stage 2

z_3	z_4	z_2^*	U_2^*
-1	-1	-1	-3
-1	1	1	-3
1	-1	-1	-3
1	1	1	-3

In terms of U_1^* we can write $f(z)$ as $f(\underline{z}) = U_1^*(z_2 \ z_3 \ z_4) + z_2(1-z_3z_4) + z_3-z_4$

Minimization of $f(\underline{z})$ with respect to z_2 gives the table for stage 2. Note that U_2^* is independent of z_3 and z_4 . Therefore $f(z) = -3+z_3-z_4$. Minimizing finally with respect to z_3 and z_4 we obtain

$f(\underline{z}) = -3$ when $z_3^* = -1$ and $z_4 = 1$ which on back substitution gives $z_2^* = -1$ and $z_1 = -1$. Therefore we obtain $(1 \ 1 \ 1 \ 0)$ as the recovered data vector.

Example 3.3 (Reed Muller Code) Consider the generator matrix given by

$$G^T = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Let $\underline{x} = (1 \ 1 \ 1 \ 1)^T$ be the transmitted vector

$$\underline{c} = (1 \ 0 \ 0 \ 1 \ 0^* \ 1 \ 1 \ 0^*) \quad \text{and} \quad \underline{r} = (1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0)^T$$

In this case the function to be minimized is given by

$$f(\underline{z}) = -\frac{1}{2} [-z_1 + z_1z_2 + z_1z_3 - z_1z_2z_3 - z_1z_4 - z_1z_2z_4 + z_1z_3z_4 + z_1z_2z_3z_4]$$

$f(\underline{z})$ can be written as

$$f(\underline{z}) = \frac{z_2}{2} (-z_1 + z_1z_3 + z_1z_4 - z_1z_3z_4) + \frac{1}{2} (z_1 - z_1z_3 + z_1z_4 - z_1z_3z_4)$$

For this case the table obtained for the first stage is shown below

stage 1

z_1	z_3	z_4	z_2^*	U_2
1	-1	-1	-1	-2
-1	-1	1	\emptyset	0
-1	1	-1	1	-1
-1	1	1	\emptyset	0
1	-1	-1	1	-2
1	-1	1	\emptyset	0
1	1	-1	\emptyset	0
1	1	1	\emptyset	0

$$U_2^*(z_1, z_3, z_4) = \min_{z_2} \frac{z_2}{2} (-z_1 + z_1 z_3 + z_1 z_4 - z_1 z_3 z_4)$$

Note that in the first stage we have considered z_2 instead of z_1 . z_1 also can be considered as the first stage variable.

From the table it is not difficult to see that $z_2^* = z_1 z_3 z_4$. Substituting z_2^* in $f(\underline{z})$ we obtain

$$f_2(z_1, z_3, z_4) = \frac{z_1}{2} (1 - z_3 + z_4 - z_3 z_4) + \frac{1}{2} (-z_3 z_4 + z_4 + z_3 - 1)$$

Table for the second stage minimization is given below

stage 2

z_3	z_4	z_1^t	U_1
-1	-1	\emptyset	0
-1	1	-1	-2
1	-1	\emptyset	0
1	1	\emptyset	0

From the table it is easy to see that $z_1 = z_3$ Substituting for z_1 in $f_2(z_1 z_3 z_4)$ we obtain

$$f_3(z_3 z_4) = z_3^{-1}$$

Therefore we obtain $z_3 = 1$ $z_4^* = \emptyset$ $z_2^t = \emptyset$ $z_1^* = -1$

where \emptyset denotes either 1 or -1 Thus the maximum likelihood solutions are

$$(-1 \ 1 \ -1 \ 1) \text{ and } (-1 \ -1 \ -1 \ 1)$$

The corresponding 0 1 vectors are (1 0 1 0) and (1 1 1 1)

Thus although the transmitted vector is (1 1 1 1) we obtain (1 0 1 0) also as a solution This is because two errors were introduced in the code vector at the places indicated by * in \underline{c} The same received vector \underline{r} can be obtained by introducing

two errors in the locations 2 and 4 of the code vector \underline{c}_1 obtained by (1 0 1 0) as shown below

$$\underline{c}_1 = (1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 0 \ 0)^T$$

Example 3.4 Consider the generator matrix given by

$$G^T = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix}$$

d_j 's can be obtained by reading columnwise in G^T and associating z_1 's whenever 1's are encountered. For instance we obtain d_{16} by noting that the last column contains 1 in the z_6 th position. Therefore $d_{16} = z_6$. The objective function is then given by

$$f(z_1, z_2, \dots, z_6) = -\frac{1}{2} \sum_{i=1}^{16} d_i r_i = -\frac{1}{2} (-z_2 - z_1 z_2 + z_1 z_3 + z_1 z_2 z_3 - z_2 z_4 - z_2 z_5 - z_4 - z_3 z_5 - z_3 z_4 z_5 - z_4 z_6 - z_4 z_5 z_6 - z_5 - z_5 z_6 + 2z_6)$$

where we assume that the transmitted vector is $(0 \ 1 \ 0 \ 0 \ 1 \ 0)^T$

$$\underline{c} = (0^* \ 0 \ 1 \ 1 \ 0 \ 1^* \ 1 \ 1 \ 1 \ 1^* \ 0 \ 1 \ 1 \ 1 \ 0 \ 0)^T$$

and $\underline{x} = (1 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0, 1 \ 1 \ 1 \ 0 \ 0)$

Note that 3 errors occur in this case

Now carrying out the steps of the recursive algorithm we obtain the following tables

Stage 1

z_2	z_3	z_1	U_1^*
1	1	1	1
1	-1	-1	-3
-1	1	1	1
-1	-1	1	-1

Stage 2

z_3	z_4	z_2	U_2^*
1	1	1	-4
1	-1	1	-2
-1	1	\emptyset	-2
-1	-1	1	-2

$$U_1 = z_1(z_2 - z_3 - z_2 z_3)$$

$$U_2 = z_2(1 + z_4 + z_3 z_4) + U_1^*$$

In stage 2 it is easy to note that $z_2^t = -z_3 z_4$ if we put $\emptyset = 1$

Stage 3

z_4	z_5	z_3^*	U_3^*
1	1	-1	-4
1	-1	1	-6
-1	1	\emptyset_1	-2
-1	-1	\emptyset_2	2

Stage 4

z_5	z_6	z_4^*	U_4^*
1	1	-1	-4
1	-1	1	-6
-1	1	1	-6
-1	-1	1	-6

$$U_3 = z_3(z_5 + z_4 z_5) + U_2^*$$

$$U_4 = 4(6 + z_5 z_6) + U_3^*$$

$$z_3^* = -z_4 z_5 \quad \text{if } \phi_1 = 1 \quad \text{and} \quad \phi_2 = -1$$

Stage 5

z_6	z_5	U_5	
1	-1	-8	$U_6 = -2z_6 + U_5^*$
-1	1	6	$U_6^* = -10$
			$z_6^* = 1$

$$U_5 = z_5(1+z_6) + U_4$$

Tracing back through the tables we obtain

$$z_6 = 1 \quad z_5 = -1 \quad z_4 = 1 \quad z_3 = 1 \quad z_2 = -1 \quad z_1^* = 1$$

Therefore the maximum likelihood solution vector is

$(0 \ 1 \ 0 \ 0 \ 1 \ 0)^T$ which is the transmitted data vector

CHAPTER 4

DECISION FEEDBACK DECONVOLUTION SCHEMES

In the previous chapter we derived a ML RDA whose computational complexity and storage requirements grow linearly with the block length and exponentially with the channel memory. We showed that attempts to further alleviate computational and storage requirements in the binary case lead to minimization of pseudo Boolean functions which in general is a rather difficult problem. The computational and storage requirements however can be reduced if one is willing to accept suboptimal schemes. Towards this end in this chapter we consider the possibility of incorporating decision feedback idea for recovering data in BDI systems. We propose and analyse four decision feedback data recovery (DF-DR) schemes. All these schemes are based centrally on the idea of nonlinear Gaussian elimination method described below.

Consider the received vector \mathbf{r} given by (2.7)

$$\mathbf{r} = \mathbf{H}\mathbf{x} + \mathbf{w} \quad (4.1)$$

Under the assumption of real valued data and white Gaussian noise the ML data vector is obtained by solving

$$\mathbf{H}^T \mathbf{H} \mathbf{x} = \mathbf{r}^T \quad (4.2)$$

One method of solving (4.2) is to use the Gaussian elimination (GE) algorithm which consists of forward elimination procedure (in which $H^T H$ is reduced to an upper triangular matrix) and back substitution (in which the solution is obtained by recursively solving the resulting set of equations). Now we can incorporate the discrete nature of data symbols into GE algorithm in the following manner. While back substitution instead of using the calculated values directly we can use the value obtained by passing the calculated value through a nonlinear decision device which assigns an appropriate value from the finite alphabet. Since this procedure involves introduction of a nonlinear device between the forward elimination and back substitution stages the method is called nonlinear Gaussian elimination.

This chapter is organized as follows. In Section 4.1 we use Cholesky decomposition technique to derive a DF-DR scheme which does not take noise variance into account. Section 4.2 gives a DF-DR scheme which minimizes mean square error between the actual data and its estimate. A numerically stable DF-DR scheme is derived in Section 4.3. This scheme is based on the minimum property of Fourier expansion in finite dimensional Hilbert spaces. In Section 4.4 a DF-DR scheme is derived by modifying Austin's [13] assumptions regarding decision feedback equalizer for CSDT systems. In Section 4.5 we show that the modified Austin's approach is also essentially equivalent to a

nonlinear Gaussian elimination method. In Section 4.6 we list some observations on the methodology used in this chapter for motivating the idea of decision feedback.

4.1 DECISION FEEDBACK SCHEME USING CHOLESKY DECOMPOSITION

Cholesky decomposition technique [74] deals with decomposition of positive definite matrices into product of nonsingular lower and upper triangular matrices. If V denotes a lower triangular matrix we can write $H^T H$ as

$$H^T H = V^T V \quad (4.3)$$

Using (4.3) (4.2) can be written as

$$V^T \underline{x} = V^{-1} H^T \underline{r}$$

This is equivalent to the forward elimination step of GE algorithm. Since V^T is an upper triangular matrix, estimate of the last element of \underline{x} is obtained as

$$\tilde{x}_m = c_m / v_{mm}$$

where c_m is the m th element of $\underline{c} = V^{-1} H^T \underline{r}$ and v_{mm} is the m th diagonal element of V^T . Now let \hat{x}_m be the value obtained by passing \tilde{x}_m through a decision device. We use \hat{x}_m instead of \tilde{x}_m for calculating \tilde{x}_{m-1} given by

$$\tilde{x}_{m-1} = (c_{m-1} - v_{m-1, m} \hat{x}_m) / v_{m-1, m-1}$$

Now \hat{x}_{m-1} will be the recovered data decided on the basis of \tilde{x}_{n-1} . Using \hat{x}_{m-1} and \tilde{x}_{n-2} is calculated. This procedure is continued till \hat{x}_1 is obtained.

In the foregoing approach last data of the block is recovered first. Here as in the case of decision feedback data recovery schemes for CSDT first data is recovered first although which data value is recovered first is immaterial in the case of BDT. For the sake of uniformity we incorporate this feature of CSDT systems in the following manner.

$H^T H$ can also be written as a product of upper and lower triangular matrices. That is

$$H^T H = U^T L \quad (4.4)$$

where U is the upper triangular matrix. One method of obtaining U is as follows.

$$\text{Let } A_m = H^T H \quad \text{and} \quad A_{m-1} = U_{m-1}^T U_{m-1}$$

Then A_m and U_m can be written as

$$A_m = \begin{bmatrix} a_{mm} & d_{n-1}^T \\ u_{m-1} & u_{n-1} \end{bmatrix} \quad \text{and} \quad U_m = \begin{bmatrix} y & e_{m-1}^T \\ 0 & U_{m-1} \end{bmatrix}$$

$$\text{Then} \quad u_m = \begin{bmatrix} \sqrt{2} e_{n-1}^T e_{n-1} & e_{m-1}^T U_{m-1}^T \\ U_{m-1} e_{n-1} & U_{n-1} U_{n-1}^T \end{bmatrix}$$

Therefore $y = (a_{mm} - \mathbf{e}_{m-1}^T \mathbf{e}_{m-1})^{1/2}$ and $\mathbf{d}_{m-1} = U_{m-1} \mathbf{e}_{m-1}$

Therefore $\mathbf{e}_{m-1} = U_{m-1}^{-1} \mathbf{d}_{m-1}$

Note that y is real because $\det(U_m U_m^T) = \det(U_{m-1})^2 y^2$
 $= \det(A_m)$

and A_m is a positive definite matrix. In deriving the recursive form for U_m we have followed the method given in [74] (see page 423)

In terms of U we have to solve

$$U^T \underline{x} = U^{-1} H^T \underline{r} \quad (15)$$

where U^T is a lower triangular matrix. The procedure of recovering data begins with the estimation of first data x_1 which is given as

$$\hat{x}_1 = z_1 / u_{11}$$

where z_1 is the first element of $\underline{z} = U^{-1} H^T \underline{r}$. x_1 is obtained by passing \hat{x}_1 through the nonlinear decision device. Then x_2 is obtained as

$$\hat{x}_2 = (z_2 - u_{21} \hat{x}_1) / u_{22}$$

We continue this procedure till \hat{x}_m is obtained. Figure 4.1 gives a block schematic of the data recovery scheme based on the procedure just now outlined. In the figure \underline{u}_1 denotes the 1th row of U^{-1} and \underline{v}_1 denotes 1th row of $U^T - D$ where $D = \text{diag}(u_{11}, u_{mm})$. Although \underline{u}_1 and \underline{v}_1 are shown stored in separate registers

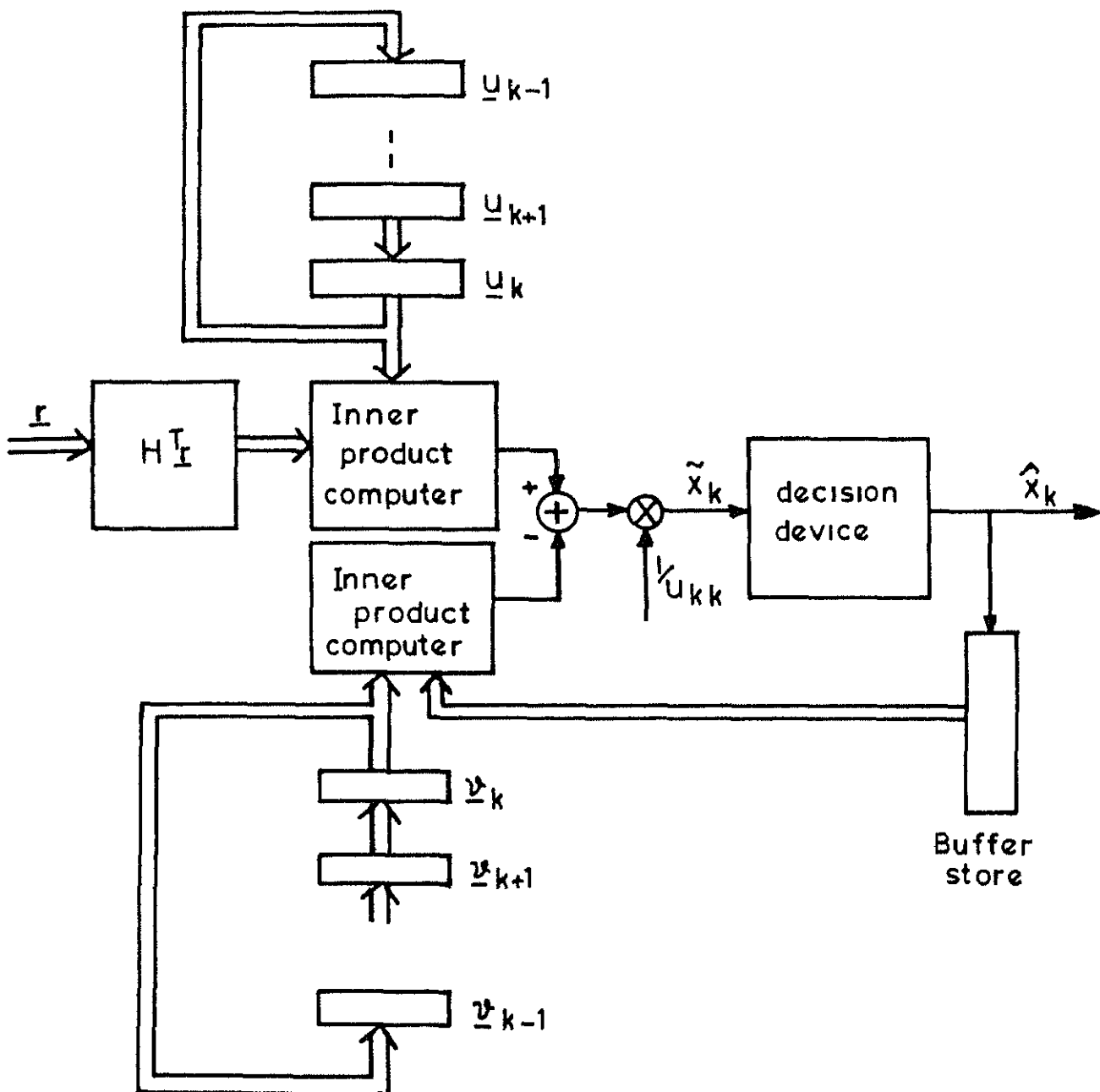


FIG 4 1 Block diagram of the data recovery scheme for BDT using Cholesky decomposition

banks in the figure in all a set of m registers of length m are sufficient

Processing of each block consists of m cycles. In the figure we have shown the status of various registers for the k th data recovery. For the next data (x_{k+1}) v_{k+1} takes the place of v_k and u_{k+1} take the place of u_k . u_{kk} becomes $u_{k+1, k+1}$. At the end of m cycles the initial state is restored and the processor is ready to deal with the next block. The inner product computer shown is the major computation block in the scheme. In the case of binary data the inner product computer used in the feedback loop does not require multiplications.

Computational requirements of DF-DP scheme for BDT using Choleky decomposition method is same as that of DF-DR scheme for CSDT if the block length is equal to the number of forward tap. On the other hand the scheme shown in Fig. 4.1 requires more storage compared to DF-JR schemes for CSDT systems. However with the present day technology storage requirement is not a crucial factor. Whereas long bursts of error in the case of CSDT due to error propagation especially in the medium SNR range is definitely a matter of concern. In the case of DF-DP scheme for BDT systems error propagation is restricted to a block length in the worst situation.

analytical evaluation of the performance of DF-DR schemes is difficult due to the presence of nonlinear decision device. However some idea of performance can be obtained under the assumption of error-free feedback. In this case probability of 1th data symbol being in error $(\text{PER})_1$ is given by

$$(\text{PER})_1 = \int_{(\text{SNR})_1}^{\infty} \frac{1}{2} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} y^2} dy = Q(\sqrt{(\text{SNR})_1}) \quad (4.6)$$

where

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-\frac{1}{2} y^2} dy$$

$(\text{SNR})_1$ can be computed as follows. From (4.5) we see that the effective noise vector is given by

$$\underline{w} = U^{-1} H^T \underline{w}$$

Therefore SNR associated with 1th data symbol is given by

$$(\text{SNR})_1 = u_{11}^2 / (||\underline{u}_1||^2 N_0/2)$$

where $||\underline{u}_1||^2$ denotes squared length of the 1th row of $U^{-1} H^T$ and $N_0/2$ is the noise variance.

4.2 WSE DECISION FEEDBACK DATA RECOVERY SCHEME

In the previous section we described one method of obtaining the parameters of DF-DR scheme using Cholesky decomposition in which we did not take noise variance into account. We now derive a DR scheme based on minimum mean square

error (MMSE) criterion. This scheme accounts for the noise variance

We first define several vectors and matrices which will be used in the following derivation

$$\underline{b} = H^T r$$

$$\underline{x}_k = (x_{k1} \ x_{k2} \ \dots \ x_{kL})^T$$

$$\underline{v}_k = (v_{k1} \ v_{k2} \ \dots \ v_{kL-k+1})^T$$

$$\underline{u}_k = (u_{k1} \ \dots \ u_{kM})^T$$

$$H_k = H \text{ with first } k-1 \text{ columns removed}$$

$$h_k = k\text{th column of } H$$

$$\tilde{H}_k = H \text{ with first } k-1 \text{ columns}$$

The objective is to obtain a set of coefficients $u_{k1} \dots u_{kM}$

$u_{k1} \dots u_{kM} \ v_{k1} \ v_{k2} \ \dots \ v_{kL-k+1}$ such that estimate of the k th data symbol given by

$$\hat{x}_k = u_{k1} b_k + \dots + u_{kM} b_M - v_{k1} x_{k1} - \dots - v_{kL-k+1} x_{kL-k+1} \quad (4.7)$$

is closest to the actual data x_k in the mean square sense

That is

$$MSE_k = E[(\hat{x}_k - x_k)^2] \text{ is minimum}$$

Here $E[\]$ denotes the standard expectation operator. Using the principle of orthogonality in MSE minimization [7] we obtain

$$E[e_k b_j] = 0 \quad j = x \quad x+1 \quad m$$

$$\text{and} \quad E[e_k x_1] = 0 \quad 1 = 1 \ 2 \quad k \ 1$$

where $e_k = x_k - \hat{x}_k$ Using the fact that $b_j = h_j^T Hx + h_j^T w$ we get

$$\begin{aligned} E[e_k b_j] &= E[(u_k^T H_k^T Hx + \underline{u}_k^T H_k^T w - \underline{v}_k^T x_k - x_k)(h_j^T Hx + h_j^T w)] \\ &= \underline{u}_k^T H_k^T H H^T h_j - \underline{v}_k^T H_k^T h_j - h_k^T h_j + \frac{N_0}{2} \underline{u}_k^T H_k^T h_j = 0 \quad (4.8) \\ j &= k \quad k+1 \quad m \end{aligned}$$

Similarly $E[e_k x_1] = 0$ gives

$$\underline{u}_k^T H_k^T h_{k-1} - \underline{v}_k^T = 0 \quad 1 = 1 \ 2 \quad k-1$$

$$\underline{v}_{k1} = \sum_{j=k}^m u_{kj} a_{j-1} \quad \text{where} \quad a_{j-1} = h_j^T h_1$$

In the matrix vector notation

$$\underline{v}_k = \tilde{H}_k^T H_k u_k \quad (4.9)$$

Substituting (4.9) in (4.8) and rearranging terms we obtain

$$\begin{aligned} \underline{u}_k &= [u_k^T [H_k^T H_k + \frac{N_0}{2} I - \tilde{H}_k^T \tilde{H}_k] H_k]^{-1} H_k^T h_k \\ &= [H_k^T H_k + \frac{N_0}{2} I]^{-1} (H_k^T H_k)^{-1} H_k^T h_k \quad (4.10) \end{aligned}$$

$$\text{But } (H_k^T H_k)^{-1} H_k^T h_k = (1 \ 0 \quad 0)^T$$

Therefore \underline{u}_k is obtained as the first column of

$$[H_k^T H_k + \frac{N_0}{2} I]^{-1}$$

Finally the resulting MSE associated with the k th data is given by

$$\begin{aligned} \text{MSE}_k &= -E[e_k x_k] = 1 - \underline{u}_k^T H_k^T h_k = 1 - \sum_{l=k}^m u_{l-k+1} \\ &= 1 - \sum_{j=0}^{\alpha} a_j u_{k-j} \end{aligned} \quad (4.11)$$

The mean square error given by (4.11) reflects to a great extent the quality of the DR scheme. It can be considered as a measure of the performance for it indicates the variance of the error. If MSE is significant probability of the k th data symbol being in error is large. Thus $1/(\text{MSE})_k$ may be considered as a measure of effective SNR associated with the k th data symbol.

Components of \underline{u}_k can be computed recursively using Trench algorithm [75] because the matrix $[H_k^T H_k + \frac{N_0}{2} I]$ is a positive definite Toeplitz matrix. First we calculate \underline{u}_m given by

$$\underline{u}_n = \underline{u}_{mm} = (H_m^T H_m + \frac{N_0}{2} I)^{-1}$$

where $H_m^T H_m = \underline{h}_m^T \underline{h}_m = h_0^2 + h_1^2 + \dots + h_g^2$

The other \underline{u}_k 's $k = n-1, m-2, \dots, 1$ can be recursively obtained by following the procedure given in [75]. Thus the computational complexity is $O(m^2)$ as the procedure for

calculating \underline{u}_1 $r = 1, 2, \dots, m$ is same as inverting a $m \times m$ positive definite Toeplitz matrix by Trench algorithm. The implementation scheme is same as the one shown in Fig. 4.1 except that \underline{u}_1 and \underline{v}_1 are given by (4.9) and (4.10) respectively.

4.3. DECISION FEEDBACK SCHEME BASED ON MINIMUM PROPERTY OF FOURIER EXPANSIONS

So far in our analysis no thought has been given to the problem of illconditioning of the channel autocorrelation matrix $H^T H$. $H^T H$ is said to be illconditioned if the ratio of maximum and minimum eigenvalue is large [74]. When $H^T H$ is illconditioned the solution obtained by inverting $H^T H$ is very sensitive to truncation errors and noise in the received vector. It is therefore essential to consider numerically stable methods. Towards this end in this section a numerically stable method of solving linear system of equations based on minimum property of Fourier expansion due to Wang [58] is adopted to derive a decision feedback data recovery scheme.

Minimum Property of Fourier Expansion For any finite or infinite linearly independent sequence of elements $\alpha_1, \alpha_2, \dots$ of a Hilbert space S_H there always exists an orthonormal system β_1, β_2, \dots such that

$$\beta_j^\dagger = \alpha_j / \|\alpha_j\| \quad j = 1, 2, \dots$$

$$\text{where } \beta_1 = \alpha_1 \quad \beta_{j+1} = \alpha_{j+1} - \sum_{k=1}^j (\alpha_{j+1} \beta_k) \beta_k / ||\beta_k||^2 \quad (4.12)$$

$$j = 1, 2$$

This procedure is known as Gram-Schmidt orthonormalization

Here $(\alpha \beta)$ denote inner product of α and β

Consider an element α in S_T . Then x_j s which minimize the norm

$$||\alpha - \sum_{j=1}^n x_j \beta_j^*||$$

are known as the Fourier coefficients of α . They are given by

$$x_j = (\alpha \beta_j^t)$$

This property of x_j s is known as the minimum property of Fourier expansion

Now let z_1, z_2, \dots, z_m be the m column vectors of H in (4.1). Since the column of H are linearly independent orthonormalizing z_1, z_2, \dots, z_m we obtain $y_1^*, y_2^*, \dots, y_m^*$. Then we have the following result due to Wang [58]

Theorem 4.1 Let r given in (4.1) be an element of an n -dimensional Hilbert space. \tilde{x}_j s which minimize the norm

$$||r - \sum_{j=1}^m x_j z_j|| \text{ are given by}$$

$$\tilde{x}_m = (r \cdot y_m) / ||y_m||^2$$

$$\tilde{x}_{m-j} = (r - \sum_{k=m-j+1}^m \tilde{x}_k z_k \cdot y_{m-j}) / ||y_{m-j}||^2 \quad (4.13)$$

$$j = 1, 2, \dots, m-1$$

(Proof of this theorem is not given in [58] Therefore for the sake of completeness we prove the theorem in the following manner)

Proof We know that given r and y_j s the c_j s which minimize the norm

$$||r - \sum_{j=1}^m c_j y_j|| \text{ are given by}$$

$$c_j = (r \cdot y_j) / ||y_j||^2$$

By substituting for y_j in terms of z_j s using (4.12) and grouping the terms common to z_j s we can obtain the recursive equations given by (4.13) To see this consider

$$\sum_{j=1}^m c_j y_j \text{ and substitute for } y_j \text{ using (4.12) This gives}$$

$$\begin{aligned} \sum_{j=1}^m c_j / y_j &= c_1 z_1 + c_2 (z_2 - \frac{(z_1 \cdot y_1) z_1}{||y_1||^2}) + c_3 (- \frac{(z_3 \cdot y_1) z_1}{||y_1||^2} \\ &\quad - \frac{(z_3 \cdot y_2) y_2}{||y_2||^2}) + \\ &\quad + c_m (z_m - \frac{(z_m \cdot y_1) z_1}{||y_1||^2} - \frac{(z_m \cdot y_{m-1}) y_{m-1}}{||y_{m-1}||^2}) \end{aligned}$$

From the above expression it is clear that the coefficient of z_m is c_m Thus

$$\hat{x}_m = c_m = (r \cdot y_m) / ||y||^2$$

The coefficient of z_{m-1} gives

$$x_{m-1} = (r - x_m z_m y_{m-1}) / ||y_{m-1}||^2$$

The rest of the coefficients can be similarly obtained

It may be noted that in the method discussed above x_m is obtained first which is used in obtaining x_{m-1} by back substitution. However for the sake of uniformity we wish to obtain x_1 first. This can be easily done by orthonormalizing the columns of H starting from the m th column. Let y_1, y_2, \dots, y_r denote the orthogonal columns obtained this way. Then the governing equations of the data recovery scheme are given by

$$\hat{x}_1 = (r - y_1) / ||y_1||^2$$

$$\hat{x}_i = (r - \sum_{k=1}^{i-1} \hat{x}_k h_k y_i) / ||y_i||^2 \quad (4.14)$$

$$i = 2, 3, \dots, m$$

where h_k is the k th column of H . $\hat{x}_1, \hat{x}_2, \dots, \hat{x}_m$ are the estimates of the data values obtained by passing $\hat{x}_1, \hat{x}_2, \dots, \hat{x}_m$ through a nonlinear decision device.

Figure 4.2 shows the data recovery scheme. Processing of each block of data consists of m cycles. Major computational load is due to the inner product computation in every cycle.

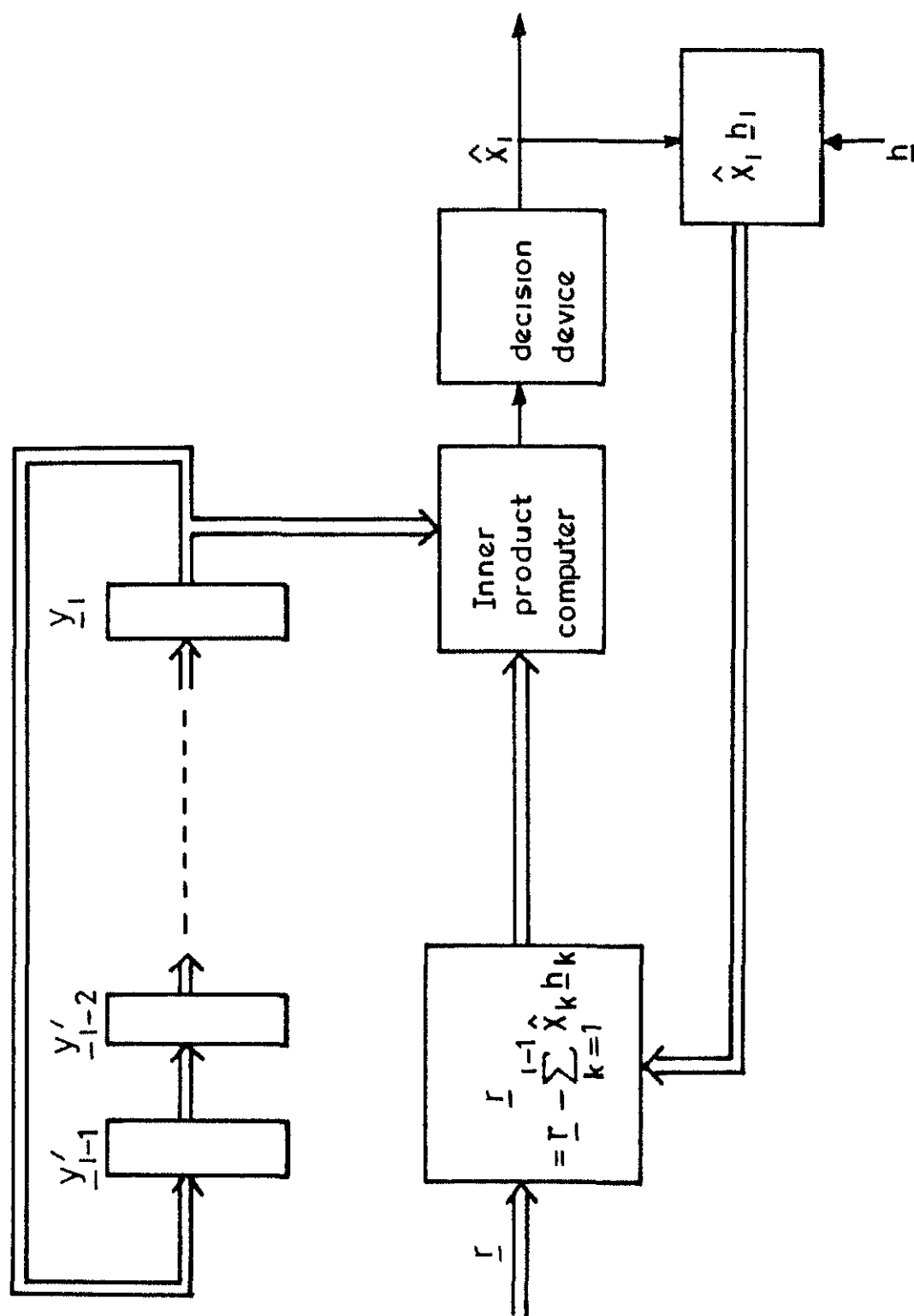


FIG 4 2 Block diagram of the data recovery scheme using minimum property of Fourier expansion

The performance of the scheme can be analytically evaluated in the absence of error. The probability of error associated with the i th data symbol is given by (4.6). The SNR associated with the i th data symbol can be calculated as follows. Substituting for r from (4.1) in (4.14) for x_1 we get

$$\begin{aligned}\tilde{x}_1 &= (Hx + w) / ||y_1||^2 \\ &= \left(\sum_{j=1}^m x_j (h_j y_1) + (w y_1) \right) / ||y_1||^2\end{aligned}$$

Substituting for h_j in terms of y_j using (4.15) we see that

$$\tilde{x}_1 = x_1 + (y_1 w) / ||y_1||^2$$

Similar expressions are obtained in the case of other data symbols also. Thus

$$(SNR)_1 = ||y_1||^2 / (N_0/2)$$

$||y_1||^2$ is thus a measure of the performance of the scheme and it is related to the channel sample response vector. When the columns of the channel convolutional matrix H are orthogonal $||y_1||^2 = 1$ for $i = 1, 2, \dots, m$ if $h^T h = 1$. When correlations exist among the columns $||y_1||^2 \leq 1$ for $i = 1, 2, \dots, m$. This can be seen by noting that

$$y_m = h_m$$

$$\text{and } y_{m-1} = h_{m-1} - \sum_{k=m-1+1}^m (h_{m-1} y_k) y_k / ||y_k||^2 \quad 1 = 1, 2, \dots, m-1$$

(4.15)

From (4.15) we obtain $||y_m||^2 = 1$ and

$$||y_{m-1}||^2 = 1 - \sum_{k=m-1+1}^m (h_{m-1} y_k)^2 / ||y_k||^2 < 1$$

4.4 DECISION FEEDBACK SCHEME BASED ON MODIFIED AUSTIN'S APPROACH

One of the most widely used DR scheme in the context of CSDT system is the decision feedback equalizer Austin [13] has derived the equations of decision feedback equalizer for CDT systems by making suitable assumptions so that detection theory results can be easily applied. As a counterpart of this scheme in this section we derive a DF-DR scheme by modifying Austin's assumptions. In the following discussion we assume binary data transmission.

Let x_k denote the symbol on which decision is to be made. We modify Austin's assumption in the following manner for the case of BDT:

1. Data symbols are independent Gaussian variables with zero-mean and unit variance for $i > k$.
2. x_1, \dots, x_k are known for $1 < k$.
3. Noise components are independent Gaussian variables with zero mean and variance $N_0/2$.
4. x_k takes 1 or -1 value with equal probability.

The likelihood ratio for x_k is given by

$$L_k = \frac{p(\underline{x}/\underline{x}_k^p \mid x_k=1)}{p(\underline{x}/\underline{x}_k^p \mid x_k=-1)}$$

where $\underline{x}_k^p = (x_1 \ x_2 \ \dots \ x_{k-1})^T$ Let $\underline{x}_k^f = (x_{k+1} \ x_{k+2} \ \dots \ x_m)^T$ denote the future data value with respect to x_k $p(\cdot)$ denotes probability density function Now L_k can be alternatively expressed as

$$L_k = \frac{\int p(\underline{x}/\underline{x}_k^p \mid x_k^f, x_k) p(\underline{x}_k^f) d\underline{x}_k^f \big|_{x_k=1}}{\int p(\underline{x}/\underline{x}_k^p \mid x_k^f, x_k) p(\underline{x}_k^f) d\underline{x}_k^f \big|_{x_k=-1}}$$

Making use of the assumptions made in the beginning of this section we can write

$$p(\underline{x}/\underline{x}_k^p \mid x_k^f, x_k) = K_1 \exp \left[\sum_{i=1}^1 x_i b_i + \sum_{i=1}^n \sum_{j=1}^n y_{ij} x_j - a_{1-j} \right] \quad (4.16)$$

where K_1 is a constant and

$$b_i = \frac{2}{N_0} \sum_{j=0}^g h_j r_{j+1} \quad i = 1 \dots m \quad (4.17)$$

$$a_i = \frac{2}{N_0} \sum_{j=0}^g h_j h_{j+1} \quad i = 0 \dots m-1 \quad (4.18)$$

Dropping all the factors which cancel out while forming L_k into K_2 we rewrite (4.16) as

$$\begin{aligned}
 p(x_k^p | x_k^f) = K_2 \exp \left[x_k b_k + \sum_{i=k+1}^m x_i b_i - \frac{1}{2} \sum_{i=1}^{k-1} \sum_{j=k+1}^m x_i x_j a_{i-j} \right. \\
 \left. - 2x_k \sum_{i=1}^{k-1} x_i a_{i-k} - 2x_k \sum_{j=k+1}^m x_j a_{k-j} \right. \\
 \left. - \sum_{i=k+1}^m \sum_{j=k+1}^m x_i x_j a_{i-j} \right]
 \end{aligned}$$

Using assumption (1) we also have

$$p(x_k^f) = K_3 \exp \left[\frac{1}{2} \sum_{i=k+1}^m \sum_{j=k+1}^m x_i x_j \delta_{ij} \right]$$

$$\text{where } \delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$$

Using the structure of multidimensional Gaussian density function and following Austin we can obtain the numerator of L_k as

$$\begin{aligned}
 \exp \left[x_k b_k - 2x_k \sum_{i=1}^{k-1} x_i a_{i-k} + \sum_{j=k+1}^m \sum_{i=k+1}^m \left(\frac{1}{2} b_j - \sum_{n=1}^{k-1} x_n a_{j-n} \right. \right. \\
 \left. \left. - x_k a_{j-k} \right) q_{j1}^k \left(\frac{1}{2} b_1 - \sum_{n=1}^{k-1} x_n a_{1-n} - x_k a_{1-k} \right) \right]
 \end{aligned}$$

where q_{j1}^k are the elements of matrix $Q_k = R_k^{-1}$. The elements of R_k are given by

$$R_{ij}^k = a_{i-j} + \frac{1}{2} \delta_{ij} \quad i, j > k \quad (4.19)$$

Thus we obtain

$$L_k = \exp[2b_k - 2 \sum_{i=1}^{k-1} x_i a_{i-k} - 2 \sum_{n=k+1}^m a_{n-k} \left[\sum_{j=k+1}^m \left(\frac{1}{2} a_j - \sum_{i=1}^{k-1} a_{i-j} x_i \right) a_{jn}^k \right]]$$

$$\text{Now let } g_{kj} = - \sum_{i=k+1}^m a_{ij}^k a_{i-k} \quad \begin{matrix} n = 1, 2, \dots, m-1 \\ j = k+1, \dots, m \end{matrix} \quad (4.20)$$

$$\text{and } f_{kj} = 2a_{j-k} + 2 \sum_{i=k+1}^m g_{ki} a_{i-j} \quad \begin{matrix} k = 2, 3, \dots, m \\ j = 1, 2, \dots, k-1 \end{matrix} \quad (4.21)$$

The decision rule in terms of g_{kj} s and f_{kj} s become

$$b_k \quad \begin{matrix} m \\ j=k+1 \end{matrix} g_{kj} b_j - \sum_{i=1}^{k-1} f_{ki} x_i \quad \begin{matrix} x_k=1 \\ \geq 0 \\ x_k=-1 \end{matrix} \quad (4.22)$$

$k = 1, 2, \dots, m$

The decision rule given by (4.22) clearly reflects the structure of the data recovery scheme. The structure is shown in Fig. 4.3 in the figure.

$$b = (b_1 \ b_2 \ \dots \ b_m)^T \quad g_k = (g_{k \ k+1} \ \dots \ g_{k \ m})^T$$

$$f_k = (f_{k1} \ f_{k2} \ \dots \ f_{k \ m-1}) \quad b_k = (b_{k+1} \ b_{k+2} \ \dots \ b_m)^T$$

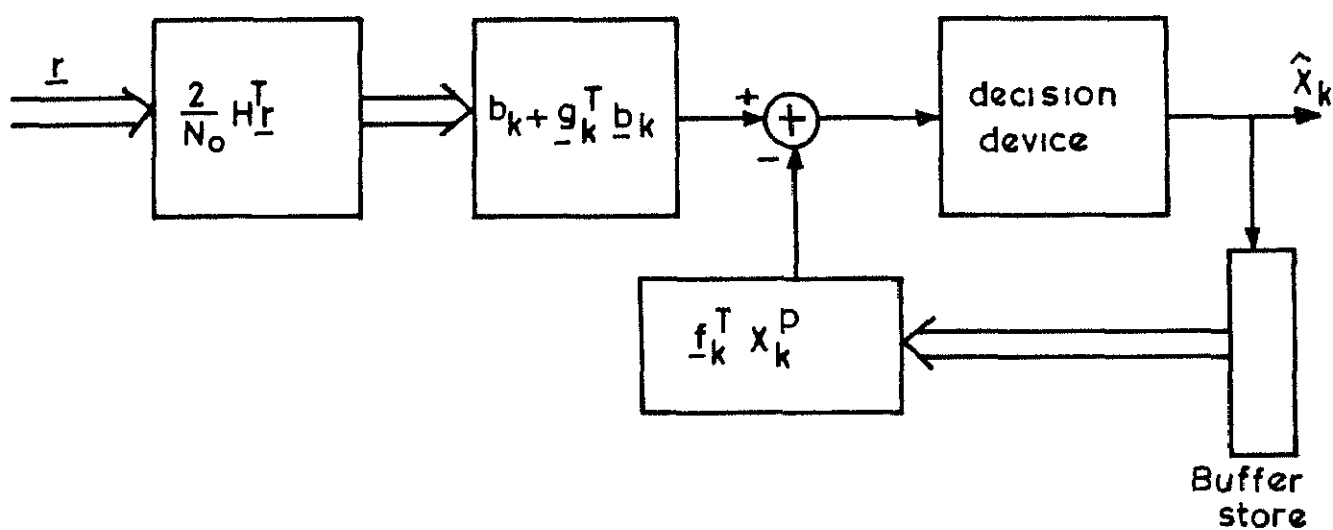


FIG 4 3 Block diagram of the data recovery scheme using modified Austins approach

In order to make decisions regarding the k th data symbol we need the knowledge of g_{lj} $j = k+1$ to n and f_{kj} $j = 1$ to $k-1$

We illustrate the foregoing discussion with an example
 Example 4.1 Consider a block containing three data values $x = (x_1 \ x_2 \ x_3)^T$ transmitted over a channel having $(h_0 \ h_1 \ h_2)$ as its finite impulse response sequence. Then for a received vector $r = (r_1 \ r_2 \ r_3)^T$ we have

$$a_0 = \frac{1}{N_0} (h_0^2 + h_1^2 + h_2^2) \quad b_1 = \frac{2}{N_0} (r_1 h_0 + r_2 h_1 + r_3 h_2)$$

$$a_1 = \frac{1}{N_0} (h_0 h_1 + h_1 h_2) \quad b_2 = \frac{2}{N_0} (r_2 h_0 + r_3 h_1 + r_4 h_2)$$

$$a_2 = \frac{1}{N_0} (h_0 h_2) \quad b_3 = \frac{2}{N_0} (r_3 h_0 + r_4 h_1 + r_5 h_2)$$

The matrices R_k $k = 1, 2$ are given by

$$R_1 = \begin{bmatrix} a_0 + \frac{1}{2} & a_1 \\ a_1 & a_0 + \frac{1}{2} \end{bmatrix} \quad R_2 = \begin{bmatrix} a_0 & \frac{1}{2} \end{bmatrix}$$

Let

$$u_1 = \begin{bmatrix} q_{22} & q_{23} \\ q_{32} & q_{33} \end{bmatrix} \quad \text{and} \quad u_2 = q_{33}$$

Then

$$g_{23} = -q_{33}a_1$$

$$f_{21} = 2a_1 + 2g_{23}a_2$$

$$g_{12} = -(q_{22}a_1 + q_{23}a_2)$$

$$f_{31} = 2a_2$$

$$g_{13} = -(q_{32}a_1 + q_{33}a_2)$$

$$f_{32} = 2a_1$$

The decision rule (4.22) becomes

$$b_1 + g_{12}b_2 + g_{13}b_3 \underset{x_1=1}{\geq} 0$$

$$b_2 + g_{23}b_3 - f_{21}x_1 \underset{x_1=-1}{\geq} 0$$

$$b_3 - f_{31}x_1 - f_{32}x_2 \underset{x_2=-1}{\geq} 0$$

$$b_3 - f_{31}x_1 - f_{32}x_2 \underset{x_3=1}{\geq} 0$$

4.1 Computation of the Tap Coefficients

The coefficients g_{kj} , $j = k+1, \dots, m$ can be viewed as the coefficients of the forward filter in the case of x_k . Moreover, these coefficients can be calculated efficiently using Levinson-Durbin algorithm [7]. For that purpose (4.20) is written below in the matrix vector form

$$\underline{g}_k = -\underline{A}_k \underline{a}_k$$

where $\underline{g}_k = (g_{k, k+1}, \dots, g_{k, m})$ and $\underline{a}_k = (a_1, a_2, \dots, a_{m-k})$. Since

Q_k is the inverse of P_k we can write \underline{g}_k as the solution of

$$P_k \underline{g}_k = -a_k$$

where R_k is defined by (4.19)

R_k is a symmetric Toeplitz matrix. Therefore we can make use of Levinson-Durbin recursive algorithm given in [7] for obtaining \underline{g}_k 's. The required steps for computing \underline{g}_k 's are given below

$$1 \quad a_{11} = a_1 / (1 + \frac{1}{2}) \quad a_0 = a_0 + \frac{1}{2}$$

$$2 \quad d_{11} = (a_1 - \sum_{j=1}^{1-1} d_{1-1-j} a_{1-j}) / e_{1-1}$$

$$3 \quad d_{1j} = d_{1-1j} - a_{11} d_{1-1-1-j} \quad 1 < j < 1-1$$

$$1 = 2, 3, \dots, k-1$$

$$4 \quad e_1 = (1 - d_{11}^2)^{1-1}$$

$$5 \quad g_{n-1, m} = -d_{11}$$

$$6 \quad g_{k, k+1} = d_{n-1-1} \quad 1 = 1, 2, \dots, m-1 \quad k = 1, 2, \dots, m-1$$

4.4.2 Modified Austin's Approach and Nonlinear Gaussian Elimination Method

In this section we show that the modified Austin's approach may be viewed as nonlinear Gaussian elimination method with appropriate modifications. For this purpose (4.22) may be written using matrix vector notation as

$$\begin{bmatrix} 1 & g_{12} & g_{13} & \dots & g_{1n} \\ 0 & 1 & g_{23} & \dots & g_{2m} \\ 0 & 0 & 1 & \dots & g_{3m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & & & & 1 \end{bmatrix} \begin{bmatrix} b \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 \\ f_{21} & 0 & 0 & \dots & 0 \\ f_{31} & f_{32} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ f_{n1} & f_{n2} & f_{nm} & \dots & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_m \end{bmatrix} \approx \underline{0}$$

$$Gb - Fx \approx 0 \quad (4.23)$$

where $\underline{0}$ is a null vector and \approx denotes symbolically the nonlinear decision device. Note that the right hand side of (4.22) is an estimate of x_k . Therefore the right hand side of (4.23) is an estimate of the data vector x . That is

$$Gb - Fx = I_m \underline{x}$$

Let $G = F + I_m$. Then the above equation can be written as

$$G^{-1} F x = b$$

(G is invertible since the diagonal elements are nonzero)

but $\underline{b} = H^T \underline{r} / (N_0/2)$ which gives

$$\frac{2}{N_0} G^{-1} F \underline{x} = H^T \underline{r} \quad (4.24)$$

Comparing (4.24) with (4.2) it can be seen that $\frac{2}{N_0} G^{-1} F$ plays the role of $H^T H$. Thus we may call $\frac{2}{N_0} G^{-1} F$ as effective autocorrelation matrix. With this interpretation we see that modified

Auslin's approach is equivalent to nonlinear Gaussian elimination method.

1.5 SOME OBSERVATIONS

As we are dealing with the problem of solving linear equations in the presence of noise when the unknown variables take values from a finite set it is natural to think that available methods for solving linear equations must apply with appropriate modifications. We demonstrated this by considering four DF-DR schemes which are centrally based on the idea of nonlinear Gaussian elimination method.

The two DF-DR schemes considered in Section 4.1 and Section 4.3 essentially modify already existing methods for solving linear equations by incorporating decision feedback as explained in the beginning of the chapter. However, the methods discussed in Sections 4.2 and 4.4 are derived based on a set of assumptions and a performance criterion. The DF-DR scheme of Section 4.2 employs mathematically tractable MSE criterion while in Section 4.4 we used more difficult error theory approach. In this approach the complexity of the scheme is alleviated considerably by assuming that the data values are independent Gaussian variables. The major underlying assumption in both the schemes is that the past decisions are error free.

In the absence of noise all the four schemes undoubtedly provide error free performance even without the nonlinear decision device. How far decision feedback helps to improve

the performance in the presence of noise can only be evaluated with the help of simulation. Although decision feedback is an old idea [77] no useful mathematical analysis is available which deals with analytical evaluation of performance. The major stumbling block appears to be the nonlinear decision device which defies any analysis. In Chapter 2 we give simulation results on DF-DF schemes developed in sections 4.3 and 4.4.

CHAPTER 5

DATA RECOVERY SCHEMES USING k -CIRCULANT COMPLETIONS

In the previous chapter we made use of past decisions and band-limited Toeplitz nature of the channel autocorrelation matrix $H^T H$ for deriving decision feedback deconvolution schemes which are suboptimal compared to the ideal deconvolution scheme of Chapter 3. It may be observed that so far in our study no attention has been given to the convolution matrix H of the channel. We shall see in this chapter that H has some interesting properties compared to $H^T H$ from the point of view of reducing computations and storage requirements. Based on k -circulant completion of P we develop two data recovery schemes. These schemes are easily implementable and have reduced computational complexity. Furthermore, unlike the DF-DP schemes of Chapter 4 which are nonlinear, the DR schemes studied in this chapter are linear and they permit analytical evaluation of error performance.

The chapter is organized as follows. In Section 5.1 we introduce the notion of k -circulant completion and describe its application to deconvolution. Two methods of inverting k -circulant matrices are given in Section 5.2. The first method uses FFT for inverting k -circulant completion of H . Complexity

of this algorithm is $O(n \log_2 n)$ where n is the number of rows of H . The second method makes use of periodicity and block k -circulant nature of the r -Circulant completion of H to provide an $O(n \log_2 g_1)$ algorithm where $g_1 = g_1$. In Section 5.3 we discuss two DP schemes which are based on k -DFT domain deconvolution discussed in Section 5.1 and direct inversion of r -Circulant completion of H described in Section 5.2. In Section 5.4 we obtain explicit expression for probability of error and compare it with the performance of DP scheme based on least squared convolution method.

5.1 r -CIRCULANT COMPLETION AND FINITE DECONVOLUTION

Consider the received vector \underline{r} given by (2.7)

$$\underline{r} = H\underline{x} + \underline{w} \quad (5.1)$$

When no statistical characterization of \underline{x} and \underline{w} are available we know that the maximum likelihood estimate of \underline{x} under the assumption of real valued data is given by

$$\underline{x}_L = (H^T H)^{-1} H^T \underline{r}$$

The most time consuming operation in obtaining \underline{x}_{LS} is the inversion of $H^T H$. Even if we assume that $(H^T H)^{-1} H^T$ is computed before the transmission of data memory requirements can become prohibitive. Therefore even with the availability of several fast algorithms for inverting $H^T H$ [78] [80] there is

still need or processing algorithms that are easily implementable. In this section we consider the application of k -Circulant matrices [81] [82] for solving deconvolution problems. For the sake of easy reference some of the results regarding k -Circulant matrices are given in Appendix D.

As an illustration we consider (5.1) and its 1-Circulant counterpart for $n = 5$, $m = 3$ and $g = 2$. In this case (5.1) can be written as

$$\begin{bmatrix} r_0 \\ r_1 \\ r_2 \\ r_3 \\ r_4 \end{bmatrix} = \begin{bmatrix} h_0 & 0 & 0 \\ h_1 & h_0 & 0 \\ h_2 & h_1 & h_0 \\ 0 & h_2 & h_1 \\ 0 & 0 & h_2 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix} = \begin{bmatrix} h_0 & 0 & 0 & kh_2 & kh_1 \\ h_1 & h_0 & 0 & 0 & kh_2 \\ h_2 & h_1 & h_0 & 0 & 0 \\ 0 & h_2 & h_1 & h_0 & 0 \\ 0 & 0 & h_2 & h_1 & h_0 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix}$$

$$\underline{r} = H_1 \underline{x} + \underline{w} \quad (5.2)$$

where $H_{k,1}$ is the k -Circulant completion of H and x_e is obtained by padding x with two zeros. It is not difficult to see that the representation given in (5.2) in terms of H_k is also valid in the general case when r is finite ($n = m+g$) and x is padded with $n-m$ zeros to obtain x_e .

From now on we call H_1 the Circulant completion of H as it is easy to see from (5.2) that H_1 is a Circulant matrix. Similarly H_{-1} will denote the Skew Circulant completion of H .

5.1.1 Deconvolution Using Circulant Completion

Consider the circulant completion of \underline{H} . If $\underline{P} = \underline{X} * \underline{Y}$ denote DFT of \underline{x} and \underline{w} respectively using the property of cyclic convolution [59] we can write (5.2) as

$$\underline{R} = \underline{H} \underline{X} + \underline{W} \quad (5.3)$$

where \underline{H} denotes the DFT of the first column of \underline{H} ($\underline{H} \underline{X}$) denotes point wise multiplication of the components of the individual vectors. Equation (5.3) suggests an easy way of deconvolving whenever none of the components of \underline{H} is too small or zero. In such cases an estimate of \underline{x} can be obtained as

$$\underline{x}_{le} = F^{-1}(\underline{R}/\underline{H}) \quad (5.4)$$

where F denotes the DFT matrix of order n and $(\underline{R}/\underline{H})$ is a vector obtained by componentwise division. This method of deconvolution can be attributed to Cooley [83] who after becoming instrumental in the development of FFT algorithm [84] proposed the foregoing approach as an application of FFT in the context of signal processing. Some of the problems associated with such straightforward application to situations involving erroneous measurements have been discussed at length by Hunt [59].

5.1.2 Deconvolution Using k-Circulant Completion

In this section we show that (5.3) and (5.4) can also be generalized to the case of k -Circulant completions. For this

purpose let $\underline{b} = (b_0 \ b_1 \ \dots \ b_{n-1})^T$ and $\underline{c} = (c_0 \ c_1 \ \dots \ c_{n-1})^T$ be two vectors. We define k -cyclic convolution of \underline{b} and \underline{c} as another vector \underline{d} given by

$$d_1 = \sum_{j=0}^{n-1} b_j \cdot c_{(1-j) \bmod n} \quad (5.5)$$

We define k -DFT of \underline{b} and \underline{c} as (see Appendix B)

$$B_{k1} = \sum_{j=0}^{n-1} b_j (\lambda w^1)^j \quad 1 = 0 \ 1 \dots n-1 \quad (5.6)$$

$$C_{k1} = \sum_{j=0}^{n-1} c_j (\lambda w^1)^j \quad 1 = 0 \ 1 \dots n-1 \quad (5.7)$$

respectively. The corresponding inverse transforms are

$$b_1 = \frac{1}{n} \sum_{j=0}^{n-1} B_{kj} (\lambda w^j)^{-1} \quad 1 = 0 \ 1 \dots n-1 \quad (5.8)$$

$$c_1 = \frac{1}{n} \sum_{j=0}^{n-1} C_{kj} (\lambda w^j)^{-1} \quad 1 = 0 \ 1 \dots n-1 \quad (5.9)$$

where λ is an n th root of 1 and $w = e^{j2\pi/n}$. Now we prove the following theorem.

Theorem 5.1 If B_k and C_k represent k -DFT of \underline{b} and \underline{c} respectively given by (5.6) and (5.7)

$$d_1 = \frac{1}{n} \sum_{j=0}^{n-1} B_{kj} C_{1j} (\lambda w^j)^{-1} \quad 1 = 0 \ 1 \dots n-1 \quad (5.10)$$

$$\text{or} \quad \underline{D}_k = (\underline{B}_k \ \underline{C}_k) \quad (5.11)$$

where \underline{D}_k is the k -DFT of \underline{d}

Proof Substituting for b_j and c_j from (5.8) and (5.9) in (5.10) we obtain

$$d_1 = \sum_{j=0}^{n-1} \sum_{t=0}^{n-1} b_{kt} (\lambda w^t)^{-j} \sum_{s=0}^{n-1} c_{ks} (\lambda w^s)^{(1-j)}$$

$$= \sum_{j=1}^{n-1} \sum_{t=0}^{n-1} b_{kt} (\lambda w^t)^{-j} \sum_{s=0}^{n-1} c_{ks} (\lambda w^s)^{-(1-j+n)}$$

Since $\lambda^n = 1$ on rearranging the above expression becomes

$$d_1 = \frac{1}{n} \sum_{j=0}^{n-1} \sum_{t=0}^{n-1} \sum_{s=0}^{n-1} b_{kt} c_{ks} \lambda^{-1} w^{-s} w^j (s-t)$$

$$+ \frac{1}{n} \sum_{j=1}^{n-1} \sum_{t=0}^{n-1} \sum_{s=0}^{n-1} b_{kt} c_{ks} \lambda^{-1} w^{s1} w^j (s-t)$$

Changing the order of summation it becomes

$$d_1 = \frac{1}{n} \sum_{t=0}^{n-1} \sum_{s=0}^{n-1} b_{kt} c_{ks} \sum_{j=0}^{n-1} \lambda^{-1} w^{-s1} w^j (s-t)$$

$$\text{But } \sum_{j=0}^{n-1} w^j (s-t) = \begin{cases} n & \text{if } s=t \\ 0 & \text{if } s \neq t \end{cases}$$

$$\text{Therefore } d_1 = \frac{1}{n} \sum_{s=0}^{n-1} b_{ks} c_{ks} (\lambda w^s)^{-1} \quad 1 = 0 \quad 1 \quad n-1$$

*

Using (5.11) we easily see that if H_k , γ_k and w_k represent k-DFT of h , \underline{x} and w respectively then

$$R_k = H_k \gamma_k + v_k$$

From which the deconvolution solution can be obtained as

$$\underline{x}_k = F_k^{-1} (R_k / \underline{H}_k) \quad (5.12)$$

provided none of the components of \underline{H}_k are zero

5.1.3 Need for Deconvolution by k-Circulant Completion

Note that when some of the DFT coefficients of the channel are zero (4) cannot be used for deconvolution as division by zero is not permitted. In such cases even if H_1 is singular for some other k , H_k may be invertible (i.e., all the k -DFT coefficients are nonzero). Thus in some cases it may be more advantageous to use k -DFT instead of 1-DFT. As an illustration consider a simple example of a channel having the following H matrix when $n = 4$ and $m = 3$

$$H = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \quad H_1 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \quad H_{-1} = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

It is not difficult to see that H_1 (Circulant completion) is singular whereas H_{-1} the Skew Circulant completion of H is nonsingular. In fact $\det(H_k)$ for this channel is 1- k . Therefore H_k is nonsingular for $k \neq 1$.

It may be further noted that computation of k -DFT is as simple as 1-DFT computation by FFT. To see this consider

the DFT of b_j given by (5.6). It can be alternatively written as

$$B_{k1} = \sum_{j=0}^{n-1} (b_j \lambda^j) w^{1j}$$

where λ is an n th root of ϵ . Computation of B_{k1} 's may be carried out by computing FFT of $b_j \lambda^j$'s $j = 0, 1, \dots, n-1$ instead of b_j 's. This aspect makes the DIT application more attractive.

5.2 DECONVOLUTION BY DIRECT INVERSE

The estimate x_{1e} or x_k obtained via 1-DFT and k -DFT given by (5.4) and (5.12) can also be obtained by directly inverting H_k whenever H_k is nonsingular. Therefore we can write

$$\underline{x}_{ke} = H_k^{-1} r \quad (5.13)$$

We now present two methods of inverting H_k .

Method 1

H_k can be expressed as (see Appendix B)

$$H_k = D^{-1} C D$$

where D is a diagonal matrix with $1, \lambda, \lambda^2, \dots, \lambda^{n-1}$ as the diagonal entries. C is a Circulant matrix with first column elements Dh where h is the channel impulse response vector. Therefore C^{-1} can be obtained by using FFT. Then H_k^{-1} is given by

$$H_k^{-1} = D^{-1} C^{-1} D$$

when n is a power of 2 inversion of C can be carried out using $2n \log_2 n + n$ complex arithmetic operations [82]. Therefore the complexity of this method is $O(n \log_2 n)$.

Elements of the first column of H_k^{-1} are obtained as follows. Let C_0, C_1, \dots, C_{n-1} denote first column elements of C^{-1} . If C_0, C_1, \dots, C_{n-1} denotes the first column elements of H_k^{-1} then C_1, \dots, C_{n-1} are given by $C_1 = C_1 / \lambda^{-1}$, $i = 0, 1, \dots, n-1$.

Method 2

In this method sparse nature of H_k is exploited to derive an $O(n \log_2 g_1)$ complexity algorithm for inverting H_k^{-1} . First we illustrate the method for Circulant matrices when $n = 2g_1$ where $g_1 = g+1$. Let $n = 8$ and $g = 3$. Then the Circulant matrix has the following form

$$H_1 = \begin{bmatrix} h_0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 \\ h_1 & h_0 & 0 & 0 & 0 & 0 & h_3 & h_2 \\ h_2 & h_1 & h_0 & 0 & 0 & 0 & 0 & h_3 \\ h_3 & h_2 & h_1 & h_0 & 0 & 0 & 0 & 0 \\ 0 & h_3 & h_2 & h_1 & h_0 & 0 & 0 & 0 \\ 0 & 0 & h_3 & h_2 & h_1 & h_0 & 0 & 0 \\ 0 & 0 & 0 & h_3 & h_2 & h_1 & h_0 & 0 \\ 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & h_0 \end{bmatrix} = \begin{bmatrix} P & Q \\ Q & P \end{bmatrix}$$

Note that H_1 is a sparse matrix. Direct application of FFT means computation of DFT of the first column having 4 zeros. Another more efficient approach is to write H_1^{-1} as

$$H_1^{-1} = \begin{bmatrix} A & B \\ B & A \end{bmatrix}$$

Whenever H_1^{-1} exists $H_1^{-1}H_1 = I_n$. Thus we obtain

$$PA + QB = I_{g_1}$$

(5.14)

$$PB + QA = O_{g_1}$$

Solving the two equations we obtain

$$A+B = (P+Q)^{-1} \quad \text{and} \quad A-B = (P-Q)^{-1}$$

where

$$P+Q = \begin{bmatrix} h_0 & h_3 & h_2 & h_1 \\ h_1 & h_0 & h_3 & h_2 \\ h_2 & h_1 & h_0 & h_3 \\ h_3 & h_2 & h_1 & h_0 \end{bmatrix} \quad P-Q = \begin{bmatrix} h_0 & -h_3 & -h_2 & -h_1 \\ h_1 & h_0 & h_3 & -h_2 \\ h_2 & h_1 & h_0 & -h_3 \\ h_3 & h_2 & h_1 & h_0 \end{bmatrix}$$

Clearly, $P+Q$ is a circulant while $P-Q$ is a Skew Circulant. Note that both are full matrices (no zero entries). If g_1 is a power of 2 we can invert $(P+Q)$ and $(P-Q)$ using $O(n \log_2 g_1)$ computations.

Now let us consider the general case where g_1 divides n such that $r = rg_1$ ($r > 2$). We can partition H_1 in this case as

$$H_1 = \begin{bmatrix} A_0 & 0 & 0 & 0 & A_1 \\ A_1 & I_0 & 0 & 0 & 0 \\ 0 & A_1 & A_0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & A_1 & I_0 \end{bmatrix}$$

Let H_1^{-1} exist and

$$H_1^{-1} = \begin{bmatrix} B_0 & B_{r-1} & B_1 \\ B_1 & B_0 & B_2 \\ B_{r-1} & B_{r-2} & B_0 \end{bmatrix}$$

Using the identity $H_1^{-1}H_1 = I_n$ we obtain

$$\begin{aligned} A_0 B_0 + A_1 B_{r-1} &= I_{g_1} \\ A_0 B_{r-1} + A_1 B_{r-2} &= 0_{g_1} \\ A_0 B_{r-2} + A_1 B_{r-3} &= 0_{g_1} \\ A_0 B_1 + A_1 B_0 &= 0_{g_1} \end{aligned} \tag{5.15}$$

By adding the above equations we obtain

$$(B_0 + B_1 + \dots + B_{r-1}) = (A_0 + A_1 + \dots + A_{r-1})^{-1} \quad (5.16)$$

Now consider r r th roots of unity. They are

$$w^l, \quad l = 0, 1, \dots, r-1 \quad \text{where } w = e^{j2\pi/r}$$

Multiplying l th equation in (5.14) by w^l and adding we can obtain

$$(B_0 + w^{r-1}B_1 + \dots + wB_{r-1}) = (A_0 + w^{r-1}A_1 + \dots + wA_{r-1})^{-1} \quad (5.17)$$

Similarly multiplying l th equation in (5.14) by w^{lk} and adding we get the general relationship

$$\sum_{l=0}^{r-1} B_l w^{lk} = (A_0 + w^{r-k}A_{r-1} + \dots + wA_1)^{-1} \quad k = 0, 1, \dots, r-1 \quad (5.18)$$

which contains (5.16) and (5.17) as particular cases

Thus in order to determine B_l 's we have to form r k Circulants (k will be different for each Circulant) whose inversion can be achieved using $O(n \log_2 g_1)$ computations when g_1 is a power of 2. Finally B_l 's are obtained using (5.18) with $O(g_1 \log_2 g_1)$ more computations. Thus the total complexity of the method is $O(n \log_2 g_1)$ which grows linearly with n for fixed g_1 .

Extension to the case of H_k^{-1} is straightforward. In this case for $n = rg_1$ we can write

$$H_k = \begin{bmatrix} A_0 & 0 & 0 & 0 & kA_1 \\ A_1 & A_0 & 0 & 0 & 0 \\ 0 & A_1 & A_0 & & 0 \\ & 0 & 0 & A_1 & A_0 \end{bmatrix}$$

Since inverse of a k -Circulant is also a k -Circulant we can partition H_k^{-1} as

$$H_k^{-1} = \begin{bmatrix} B_0 & kB_{r-1} & kB_1 \\ B_1 & B_0 & kB_2 \\ B_{r-1} & B_{r-2} & B_0 \end{bmatrix}$$

Thus equation (5.14) becomes

$$A_0 B_0 + kA_1 B_{r-1} = I_{g_1}$$

$$A_0 B_{r-1} + A_1 B_{r-2} = 0_{g_1}$$

(5.19)

$$A_0 B_1 + A_1 B_0 = 0_{g_1}$$

Now let λ be an r th root of k . Then multiplying i th equation in (5.19) $0 < i < r-1$ by $(\lambda w)^{1J}$ and adding we obtain

$$\sum_{j=0}^{r-1} B_1 w^{1J} \lambda^1 = \left(\sum_{j=0}^{r-1} (\lambda w)^{r-j} A_1 \right)^{-1} \quad j = 0, 1, 2, \dots, r-1 \quad (5.20)$$

It may be noted that the procedure is similar to the case of Circulant matrix inversion.

5.3 DATA RECOVERY SCHEMES BASED ON k -CIRCULANT COMPLETIONS

When SNR is high the 1-DFT approach described in Section 5.1.2 can be used for recovering data. Such a scheme is shown in Figure 5.1. Although two FFT processors are shown in the figure one is sufficient in actual practice for at a time only one of them will be in use.

Note that in the case of the DF scheme shown in Fig. 5.1 for every block of data transmitted k -DFT of the received vector and an inverse DFT is to be performed. One way of avoiding calculation of DFT for every block separately is to use the elements H_k^{-1} directly in the DR scheme. Such a DR scheme is shown in Fig. 5.2. This scheme involves a circulating shift register of length n in which first row elements of H_k^{-1} are stored and circulated. The tap coefficients of the cyclic transversal filter are the received observations r_0, r_1, \dots, r_{n-1} . The multiplier with coefficient k in the feedback loop of the circulating filter depends upon the completion used.

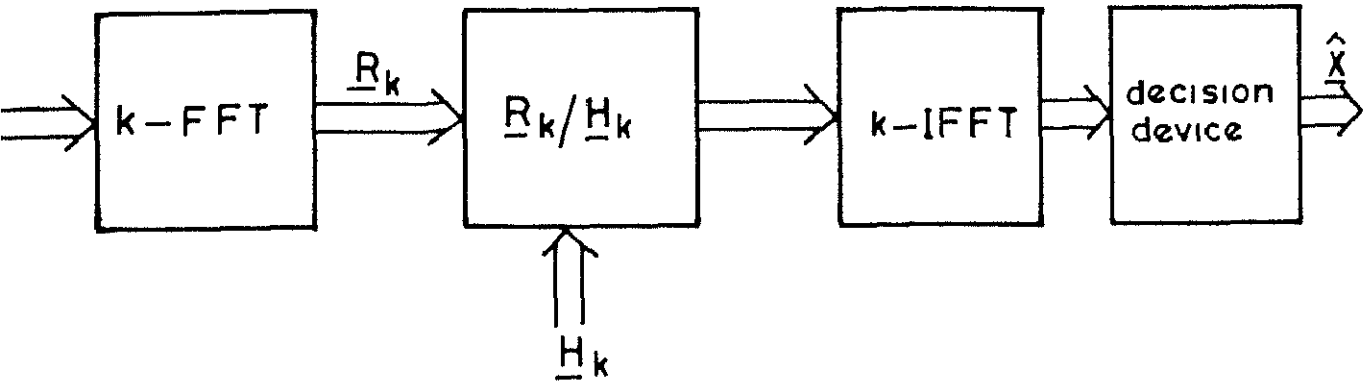


FIG 5 1 Block diagram of the data recovery scheme using $k\text{-DFT}$

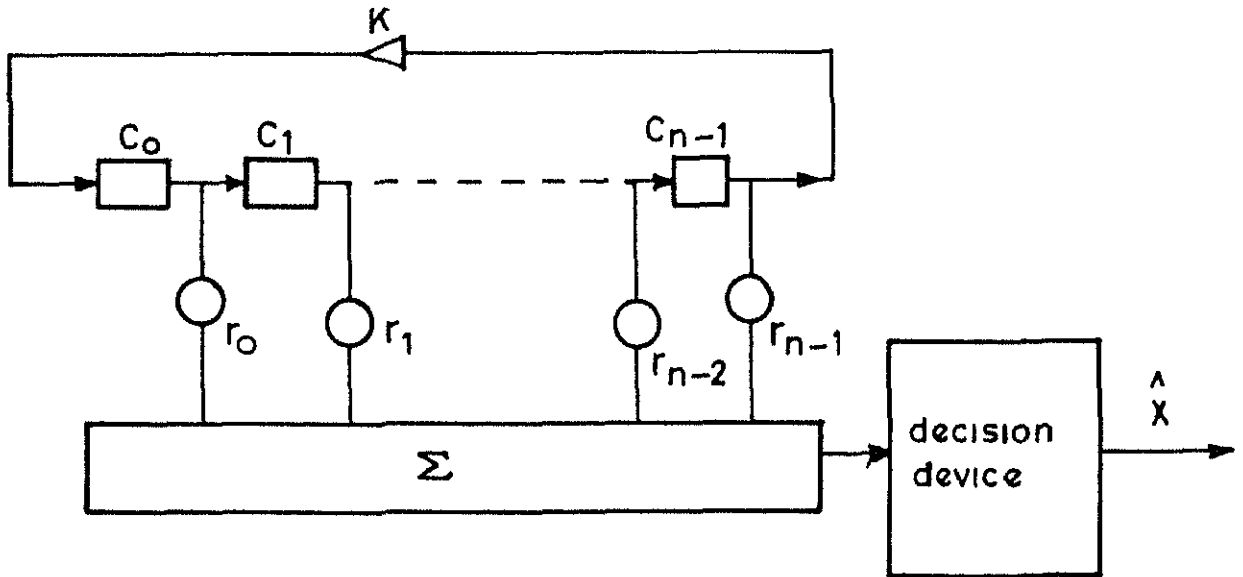


FIG 5 2 Data recovery scheme based on $k\text{-Circulant}$ completion

Processing of each data block consists of m cycles. If the entire processing is completed in nT seconds where T is the symbol duration, there will be no need for long buffers to store the incoming blocks. In this case the delay involved will be one block duration. Thus two registers of length n are sufficient. One is used in the processing while the other one stores observations arriving due to the next block.

The complexity of the DP scheme shown in Fig. 5.1 depends on the method used for implementing FFT. Since FFT algorithm is parallel and recursive it is suitable for those real time applications where cost of the DR scheme is more important than the probability of error performance.

major advantage of the DR scheme shown in Fig. 5.2 is that in its implementation complex arithmetic is totally avoided. Moreover its implementation is simple compared to DR schemes considered in the previous chapters.

5.4 PERFORMANCE CONSIDERATIONS

In the context of data transmission it has generally been observed that the complexity of DR schemes seem to be directly proportional to the desired performance level. Thus minimum probability of error (the best performance) demands highly complex DR schemes. Therefore we may infer that DR schemes based on k-Circular correlation being less complex

the r performance may not be satisfactory compared to DP schemes of chapters 3 and 4. However the DP schemes considered in this chapter have a distinct advantage over DR schemes developed in the previous chapters when the SNR is high. In this section we obtain explicit expressions for the probability of error and compare it with the performance of DR scheme based on least square deconvolution. Simulation studies on the performance of the DP schemes studied in this chapter are presented in Chapter 8.

To begin with let $k = 1$. Substituting for r from (5.1) in (5.1) we obtain for $k = 1$

$$\tilde{x}_{1e} = x_e + H_1^{-1} \underline{w} \quad (5.21)$$

From (5.21) we see that the effective noise variance depends on the elements of H_1^{-1} . Since the elements in any row is a cyclically shifted version of the first row and since noise components are statistically independent and are of equal variances the effective variance σ_1^2 is given by $\sigma_1^2 = \sigma^2 d_1^2$ where

$$d_1^2 = \sum_{i=1}^n h_{1i}^2 \quad (5.22)$$

and \underline{h}_1 is the first column of H_1^{-1} . With the knowledge of the DFT coefficients of \underline{h} we can express d_1^2 alternatively as

$$d_1^2 = \frac{1}{n} \sum_{i=1}^n \left(\frac{1}{\lambda_1 \lambda_{n-i+2}} \right) \quad \lambda_{n+1} = \lambda_1 \quad (5.23)$$

Using (5 25) (5 26) and (5 27) v obtain

$$n < n^2 \left(\sum_{i=1}^n h_{11}^2 \right) \left(\sum_{i=1}^g h_1^2 \right) \quad \text{or}$$

$$\sum_{i=1}^n h_{11}^2 > \frac{1}{n \left(\sum_{i=1}^g h_1^2 \right)}$$

Using (5 22) in the above inequality we obtain (5 24)

(5 22) and (5 24) are also valid in the case of k Circulant completion since $|k| = 1$ Therefore we can write

$$d_k^2 = \sum_{i=1}^n |h_{k1}|^2 > \frac{1}{n \left(\sum_{i=1}^g h_1^2 \right)} \quad (5 28)$$

It is interesting to note that all the data symbols recovered have the same probability of being in error because SNR associated with each of them is same It is given by

$$(\text{SNR})_1 = \frac{1}{d_k^2 \frac{N_o}{2}} \quad i = 0 \ 1 \quad m-1 \quad (5 29)$$

When $|k| = 1$ The probability of error (PER) is given by

$$\text{PER} = \frac{1}{d_k \sqrt{(N_o/2)}} \int_0^\infty \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} y^2} dy = Q\left(\frac{1}{d_k \sqrt{(N_o/2)}}\right) \quad (5 30)$$

Let us now consider the probability of error performance of the DR scheme based on LS-deconvolution solution given by

(5.2) In this case SNR associated with data symbols will be different. The average probability of error is given by

$$\overline{\text{PER}}_{\text{LS}} = \frac{1}{m} \sum_{i=1}^m Q\left(\sqrt{\frac{d_{L1}^2}{2 N_0}}\right) \quad (5.31)$$

where $d_{L1}^2 \frac{N_0}{2}$ is the variance of the noise component associated with i th data symbol. An upper bound for PER_{LS} can be obtained as

$$\text{PER}_{\text{LS}} < Q\left(\sqrt{\frac{2}{d_{\max}^2 N_0}}\right) \quad (5.32)$$

where $d_{\max}^2 = \max_{1 \leq i \leq m} d_{L1}^2$

We note that d_{L1}^2 is the squared sum of the elements of the i th row of $(H^T H)^{-1} H^T$. Equivalently they are the diagonal elements of $(H^T H)^{-1}$ (because $(H^T H)^{-1} H^T H (H^T H)^{-1} = (H^T H)^{-1}$). We now show that

$$d_k^2 \geq d_{L1}^2 \quad i = 1, 2, \dots, m \quad (5.33)$$

as a particular case of a more general result. Let

$H_g = [H \ H_c]$ where H_g is the square completion of H such that H_g is invertible. (Such completions are treated in greater detail in Chapter 6)

Let $d_{g1}^2 = 1 \dots n$ be the diagonal elements of $(H_g^T H_g)^{-1}$. In the following theorem we prove that

$$d_{g1}^2 \geq d_{L1}^2 \quad (5.34)$$

It is easy to see that (5.33) follows as a special case of (5.34) when $|k| = 1$.

Theorem 5.3 Given H_g , d_{g1} , d_{L1} as defined above

$$d_{g1}^2 \geq d_{L1}^2 \quad 1 = 1 \dots m$$

Proof $(H_g^T H_g)^{-1}$ can be written as

$$(H_g^T H_g)^{-1} = \begin{bmatrix} H^T H & H^T H_c \\ H_c^T H & H_c^T H_c \end{bmatrix}^{-1} = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix}$$

Since

$$\begin{bmatrix} H^T H & H^T H_c \\ H_c^T H & H_c^T H_c \end{bmatrix} \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} = \begin{bmatrix} I_m & 0 \\ 0 & I_{n-m} \end{bmatrix}$$

We have

$$H^T H G_{11} + H^T H_c G_{21} = I_m \quad (5.35)$$

$$H^T H G_{12} + H^T H_c G_{22} = 0_{m \times n-m} \quad (5.36)$$

Therefore from (5.36) we obtain

$$H^T H_c = -H^T H G_{12} G_{22}^{-1}$$

where G_{22}^{-1} exists because $H_g^T H_g$ is a positive definite matrix substituting for $H^T H_c$ in (35) we get

$$(H^T H)^{-1} = G_{11} - G_{12} G_{22}^{-1} G_{21} \quad (5.37)$$

$G_{21} = G_{12}^T$ as $H_g^T H_g$ is a symmetric matrix

Now let \tilde{d}_1^2 be the diagonal elements of $G_{12} G_{22}^{-1} G_{21}$

Since $G_{21} = G_{12}^T$ and G_{22} is a p.d. matrix

$G_{12} G_{22}^{-1} G_{21} > 0$ (by the definition of a p.d. matrix A
 $x^T A x > 0$ for all $x \in \mathbb{R}^n$ except $x = 0$)

which shows that \tilde{d}_1^2 is strictly positive. Now using (5.37)

$$d_{L1}^2 = d_{g1}^2 - \tilde{d}_1^2$$

where d_{g1}^2 are the diagonal entries of G_{11}

Summarizing in this chapter we have studied the application of k-Circulant completions for data recovery in the context of BDT. We have analysed two DR schemes which facilitate good reduction in computational complexity and storage requirements compared to the DR schemes studied in the previous chapter. Moreover, these schemes are amenable to performance analysis. We have obtained explicit expressions for the probability of error and compared it with the performance achievable by

DR scheme based on LS-deconvolution. In general the performance of DR schemes based on k -Circulant completions is inferior to the DR schemes of chapters 3 and 4. However in Chapter 8 we shall see that in the case of some channels DR schemes based on k -Circulant completion perform well compared to ML-DR and DF-DP schemes.

In the next chapter we take up a detailed study of the relationship between least squares deconvolution and deconvolution via k -Circulant completions.

CHAPTER 6

LEAST SQUARES DECONVOLUTION VIA k-CIRCULANT COMPLETIONS

We recall from our discussion in the previous chapter that a solution to the deconvolution problem via nonsingular k-Circulant completion can be expressed as

$$\underline{x}_k = H_{k2} \underline{r} \quad (6.1)$$

where $H_k = [H \ H_{k1}]$ and $H_k^{-1} = \begin{bmatrix} H_{k2} \\ H_{k3} \end{bmatrix}$. This approach is computationally attractive compared to the least squares (LS) solution given by

$$\underline{x}_{LS} = (H^T H)^{-1} H^T \underline{r} \quad (6.2)$$

because H_k^{-1} can be obtained using FFT. However, in general $\underline{x}_{LS} \neq \underline{x}_k$ as can be seen from the following simple example.

Example 6.1 Let $H = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix}$ and $\underline{r} = \begin{bmatrix} 1 & 2 \\ 0 & 5 \\ -1 \end{bmatrix}$

For $k = 1$ $H_1 = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$ $H_1^{-1} = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 \\ -1 & 1 & 1 \\ 1 & -1 & 1 \end{bmatrix}$

$$H_{12} = \frac{1}{2} \begin{bmatrix} 1 & 1 & -1 \\ -1 & 1 & 1 \end{bmatrix}$$

$$\text{But } (H^T H)^{-1} H^T = \frac{1}{3} \begin{bmatrix} 2 & 1 & -1 \\ -1 & 1 & 2 \end{bmatrix}$$

$$\text{Therefore } \underline{x}_{LS} = \begin{bmatrix} 1 & 1 \\ -1 & 2 \end{bmatrix} \neq \begin{bmatrix} 1 & 6 \\ -1 & 1 \end{bmatrix} = \underline{x}_1$$

Thus it becomes important and interesting to know conditions under which $\underline{x}_{LS} = \underline{x}_k$ because in such a case LS solution can be obtained with much less computation via k -Circulant completions.

In the present chapter we study this problem in a more general framework of square completion of rectangular matrices. Let $H_s = [H \ H_{s1}]$ be nonsingular square completion of H in general H_{s1} need not depend on H . The k -Circulant completion of H is a particular case of H_s where H_{s1} depends on H .

Let $H_s^{-1} = \begin{bmatrix} H_{s2} \\ -H_{s3} \end{bmatrix}$. Our interest is to obtain conditions under which

$$\underline{x}_{LS} = \underline{x}_s = H_{s2} \underline{r} \quad (6.3)$$

We show that $\underline{x}_{LS} = \underline{x}_s$ if and only if $r^T H_{s1} = 0_{m \times n-m}$. Such completions are called orthogonal completions. As a corollary it follows that $\underline{x}_L = \underline{x}_k$ iff $H^T H_{k1} = 0_{m \times n-m}$. Under this condition we further show that orthogonal k -Circulant completions of H are orthogonal k -Circulant matrices.

We organize the chapter in the following manner. In Section 6.1 a method of solving overdetermined set of linear equations using the notion of square completion of matrices is discussed. A brief review of least squares solution of linear system of equations and Moore-Penrose (MP) inverse of matrices is given in Section 6.2. In Section 6.3 we obtain conditions on square completions that result in P-inverse or least squares inverse (LS-inverse) of a full rank rectangular matrix. Square completions which result in MP-inverse are called orthogonal square completions. We further show that H_k is an orthogonal k -Circulant completion of H iff H_1 is an orthogonal k -Circulant. In Section 6.4 we determine channels which are amenable to LS solution via k -Circulant completions. We show that in most of the practical cases LS solution to the deconvolution problem cannot be obtained via (6.2) because in such cases k -Circulant completions do not result in orthogonal k -Circulants. In Section 6.5 we characterize orthogonal k -Circulants in the k -DFT domain. An alternate expression for LS solution using partitions of k -Circulant completions of H is obtained in Section 6.6 for the purpose of knowing the relationship between H_{k2} and the MP-inverse of H . We show that

$$H^- = H_{k2} - H_{k2} F_k (H_{k3} H_{k3}^*)^{-1} H_{k3}$$

where H^- is the Moore-Penros inverse of H .

6.1 SQUARE COMPLETION AND SOLUTION OF LINEAR EQUATIONS

Since the deconvolution problem is essentially solving a set of linear equations we first consider application of square completions for solving a system of linear equations given by

$$A\underline{x} = \underline{b} \quad (6.5)$$

where A is an $n \times m$ matrix with $n > m$. Let $\text{rank of } A(\text{rank}(A))$ be m . Let $A_s = [A \ A_{s1}]$ be a nonsingular square completion of A . Such a choice of A_{s1} is possible because A is a full rank matrix. Let \underline{x}_e be an extended vector

$$\underline{x}_e = \begin{bmatrix} \underline{x} \\ \underline{u} \end{bmatrix} \quad (6.6)$$

where \underline{u} is an $n-m$ length zero vector. In terms of A_s^{-1} we can obtain a solution to (6.5) given by

$$\tilde{\underline{x}}_e = A_s^{-1} \underline{b} \quad (6.7)$$

Note that $\tilde{\underline{x}}_e \neq \underline{x}_e$ in general. We will return to this point later. Now let

$$\tilde{\underline{x}}_e = \begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \end{bmatrix} \quad (6.8)$$

where \underline{x}_1 is a solution to (6.5) with respect to the completion on A_{s1} . Depending on the choice of A_{s1} different solutions are obtained. A natural question then demands answer at this

stage is whether all such solutions are equivalent in some sense or not. Because if the solution given by (6.6) is independent of A_{s1} it may be economical in terms of computations to choose A_{s1} such that the inversion of S involves least computational complexity. One simple result in this connection is the following

Theorem 5.1 If \underline{b} belongs to the range space of A then $\tilde{\underline{x}}_e$ given by (6.7) is independent of A_{s1} .

Proof Let $\underline{a}_0, \underline{a}_1, \dots, \underline{a}_{m-1}$ be the columns of A . Since \underline{b} is in the range space of A , $\underline{b} = c_0 \underline{a}_0 + c_1 \underline{a}_1 + \dots + c_{m-1} \underline{a}_{m-1}$ for some c_i 's. Therefore solution to (6) is given by

$$\underline{x} = (c_0 \ c_1 \ \dots \ c_{m-1})^T = \underline{c}$$

Let A_s be any nonsingular square completion of A . Let

$$A_s^{-1} = \begin{bmatrix} A_{s2} \\ A_{s3} \end{bmatrix} \quad \text{Then from (6.8) we see that}$$

$$\underline{x} = A_{s2} \underline{b} = \underline{s2} \ \underline{c}$$

Since $A_s^{-1} A_s = I_n$ we obtain $A_{s2} A = I_m$. Therefore $\underline{x}_1 = \underline{c}$ which is independent of A_{s2} .

Note that the condition on \underline{b} demanded by Theorem 5.1 cannot be guaranteed in practice if \underline{b} does not belong to the range space of A . We consider this case in Section 6.3. In the next section we give a brief review of the conventional methods available for dealing with this case.

6.2 LEAST SQUARE MINIMUM NORM SOLUTION MOORE-PENROSE GENERALIZED INVERSE

The method of solution normally used when \underline{b} is not in the range space of A is to constrain the solution space. The method of least squares is one way of constraining the solution space. We choose only those \underline{x} 's which minimize

$$\| \underline{x} - \underline{b} \|^2 = (\underline{x} - \underline{b})^T (\underline{A}\underline{x} - \underline{b}) \quad (6.9)$$

when A is of rank m it can be shown that the right hand side of (6.9) is minimized by

$$\underline{x}_L = (A^T A)^{-1} A^T \underline{b}$$

when A does not have full rank ($r(A) < m$) the method used is to further constrain the solution space by considering \underline{x} 's which minimize (6.9) and have minimum length. It can be shown that such a solution is unique and is given by [85] [87]

$$\underline{x}_{mLS} = A^+ \underline{b}$$

where A^+ is called the Moore-Penrose (MP) inverse of A [85]

A^+ satisfies the following properties

$$\begin{aligned} 1 \quad & A A^+ = A \\ 2 \quad & A^+ A A^+ = A^+ \\ 3 \quad & (A^+)^T = A A^+ \\ 4 \quad & (A^+ A)^T = A^+ A \end{aligned} \quad (6.10)$$

A matrix A^{LS} which satisfies the conditions $AA^{LS}A = A$ and $(AA^{LS})^* = AA^{LS}$ is called a least squares inverse of A [3]. When A is of full rank it can be shown that $A^{LS} = A^+$.

3 LEAST SQUARES SOLUTION VIA SQUARE COMPLETIONS

Another approach that can be taken when b is not in the range space of A is to extend the solution space. It is easy to see that square completion is a way of extending the solution space. The solution obtained by forming $A_s^{-1}b$ depends on A_{s1} which controls the solution space. In such situations it is instructive to know what kind of completions lead to least squares solution. Alternatively we are interested in A_{s1} which makes A_{s2} in $A_s^{-1} = \begin{bmatrix} A_{s1}^{-1} & A_{s2} \\ 0 & A_{s3} \end{bmatrix}$ a least squares inverse or Moore-Penrose inverse. The following lemma characterizes MP inverse of A via square completions.

Lemma 6.1 Let $A_s = \begin{bmatrix} A & A_{s1} \\ A_{s2} & A_{s3} \end{bmatrix}$ and $A_s^{-1} = \begin{bmatrix} A_{s2} \\ A_{s3} \end{bmatrix}$. Then A_{s2} is the MP inverse of A iff $(A_{s1}A_{s3})^* = A_{s1}A_{s3}$.

Proof We know that $A_s^{-1}A_s = A_sA_s^{-1} = I_n$. Therefore in terms of the partitions of A_s and A_s^{-1} the following matrix equations must be satisfied

$$A_{s2}A_{s3} = I_n \quad (6.11a)$$

$$A_{s2}A_{s1} = 0_{m \times n-m} \quad (6.11b)$$

$$A_{s3}A_{s1} = 0_{n-m \times m} \quad (6.11c)$$

$$A_{s1}A_{s1} = I_{n-m} \quad (6.11d)$$

$$A_{s2}A_{s1}A_{s3} = I_n \quad (6.11e)$$

Using (6.11a) it is easy to show that A_{s2} satisfies the first three properties of the P-inverse of A given by (6.10). The fourth property in (6.10) requires that

$$(A_{s2})^T = A_{s2}$$

From (6.11e) we can see that this condition will be satisfied iff $(A_{s1} \ A_{s3})^T = A_{s1} \ A_{s3}$ *

Using Lemma 6.1 we can prove the following theorem:

Theorem 6.2 Given a nonsingular completion of A

$A_s = [A_{s1}]$ and its inverse $s^{-1} = \begin{bmatrix} A_{s2} \\ A_{s3} \end{bmatrix}$ s^{-1} is the P-inverse of A iff $A^T A_{s1} = O_{m \times n-m}$

Proof Let A_{s2} be a P-inverse of A . Then $s^{-1} = (A^T A)^{-1} A^T$. Substituting for A_{s2} in (6.11b) we see that $A^T A_{s1} = O_{m \times n-m}$.

This proves the first part of the theorem.

Now assume that $A^T A_{s1} = O_{m \times n-m}$. Let $R(A)$ denote the range space of A . Then $R(A) \perp R(A_{s1})$. Here \perp indicates that $R(A_{s1})$ is orthogonal to $R(A)$. From (6.11b) we have $P(A_{s2})^T \perp R(A_{s1})$. Since $P(A_{s1}) \perp R(A)$ it follows that the rows of A_{s2} span $R(A)$. But columns of A also span $P(A)$. So $A_{s2} = D A^T$ for some nonsingular matrix D . Substituting for A_{s2} in (6.11c) we get $D = (A^T A)^{-1}$. This implies $A_{s2} = (A^T A)^{-1} A^T$ **

From now onward we shall refer to square completions s having the property $A^T A_{s1} = O_{m \times n-m}$ as orthogonal completions.

In the next theorem we consider the implication of Theorem 6.1 in the case of k -Circulant completions

Theorem 6.3 Given a nonsingular k -Circulant completion of H

$H_k = [H \ H_{k1}]^T$ and its inverse $H_k^{-1} = \begin{bmatrix} H_{k2} \\ H_{k3} \end{bmatrix}$ (6.12) is the MP-inverse of H iff H_{k1} is an orthogonal k -Circulant matrix

Proof From the definition of k -Circulant matrix (Appendix B) we recall that \underline{h}_1 the first column of H_1 is related to other columns of H_k through η_k which is a k -Circulant matrix with first row $(0 \ 0 \ \dots \ k)$ in other words

$$\underline{h}_1 = \eta_k^{1-1} \underline{h}_1 \quad i = 2, 3, \dots, n \quad (6.12)$$

where \underline{h}_1 is the i th column of H_k

Using Theorem 6.2 we know that H_{k2} is the MP inverse of H if $H^T H_{k1} = O_{n \times n-m}$

For this condition to hold we must have

$$\underline{h}_j^T \underline{h}_1 = 0 \quad j = 1, 2, \dots, m \quad i = m+1, \dots, n \quad (6.13)$$

In view of (6.12) (6.13) implies

$$\underline{h}_1^T (\eta_k^{j-1})^T \eta_k^{1-1} \underline{h}_1 = 0 \quad j = 1, 2, \dots, m \quad i = m+1, \dots, n$$

But $\eta_k^T \eta_k = k^2 I_n$ (see Appendix B)

Therefore we obtain $\underline{h}_1^T \eta_1^1 \underline{h}_1 = 0$

$$\text{or } \eta_1^T \eta_k^1 \underline{h}_1 = 0 \quad i = 1, 2, \dots, n-1$$

But $\eta_k^1 \underline{h}_1$ is the $i+1$ th column of H_k

therefore H_k in this case is an orthogonal k -Circulant

*

From the foregoing discussion it is clear that the square completion of rectangular matrices provide a general framework for the study of deconvolution problems. Both the last squares approach and the method based on k -Circulant completion are particular cases with appropriate choice of the completions. In the light of Theorem 6.3 in the next section we determine those channels which admit LS solution via k -Circulant completions.

6.4 CHANNELS AMENABLE TO LS SOLUTION VIA k -CIRCULANT COMPLETIONS

In this section we are interested in determining those channel convolution matrices H such that H_k is an orthogonal k -Circulant. Two cases are to be considered.

Case 1 $m > g$

In the case of block data transmission $m > g$ This is necessary because otherwise reduction in the effective data rate over the channel will be large. The form of H_k in this case is shown below

$$r_k = \begin{bmatrix} h_0 & 0 & 0 & kh_g & kh_{g-1} & kh_1 \\ r_1 & h_0 & & 0 & kh_g & kh_2 \\ & h_1 & & 0 & 0 & \\ & & 1_0 & & & \\ h_g & & h_1 & h_0 & & kh_g \\ 0 & h_g & & h_1 & h_0 & 0 \\ & 0 & & & & \\ 0 & 0 & h_g & h_{g-1} & h_{g-2} & h_1 & h_0 \end{bmatrix}$$

It is easy to see that for H_k to be an orthogonal k -Circulant the following equations are to be satisfied

$$kh_0 h_g = 0$$

$$k(h_0 h_{g-1} + h_1 h_g) = 0$$

$$k(h_0 h_1 + h_1 h_2 + \dots + h_{g-1} h_g) = 0$$

Since $h_0 \neq 0$ we obtain

$$h_g = 0 \quad h_{g-1} = 0 \quad h_1 = 0$$

This means that the only channel that admits LS solution via k -Circulant completion is the distortionless channel having $h_0 \neq 0$ and all other ISI components zero

I now consider the case $n > m+g$. This case arises in most of the deconvolution applications. For instance when FFT is to be used for inverting H . It is desirable to have n a power of 2 in such situations. Therefore when n is not a power of 2 the channel vector is padded with zeros. Another motivation for considering this case is that there may exist some $n > m+g$ such that LS solution is obtained via k -Circular completion.

In this case H_k will have a more general form as shown below

$$H_{ke} = \begin{bmatrix} H & H_{k4} \\ O & H_{k5} \end{bmatrix}$$

where O is a null matrix of order $n-m-g \times m$. Using the arguments given earlier we can easily show that H_{ke} is an orthogonal k -Circular if only when $h_0 \neq 0$ and $h_1 = 0$ $1 = 1, 2, \dots, g$.

Case 2 $m < g$

This situation arises in convolution situations where the number of unknowns is less than the number of impulse response coefficients.

In this case H_k takes the following form when $g = 4$ and $m = 2$

$$H_k = \begin{bmatrix} h_0 & 0 & kh_4 & kh_3 & kh_2 & kh_1 \\ h_1 & h_0 & 0 & kh & kh_3 & kh_2 \\ h_2 & h_1 & h_0 & 0 & kh_4 & kh_3 \\ h_3 & h_2 & h_0 & h_0 & 0 & kh_4 \\ h_4 & h_3 & h_2 & h_1 & h_0 & 0 \\ 0 & h_4 & h_3 & h_2 & h_1 & h_0 \end{bmatrix}$$

For H_1 to be an orthogonal k -Circulant must have

$$kh_0h + h_2h_0 + h_3h_1 + h_4h_2 = 0$$

$$kh_0h_3 + kh_1h_4 + h_3h_0 + h_4h_1 = 0$$

$$kh_0h_1 + kh_1h_2 + kh_2h_3 + h_3h_4 = 0$$

Any solution to the above set of equations serves as the desired channel. Thus for $m < g$ we see that it may be possible to obtain LS solution via k -Circulant completion. Explicit form of h_1 's is difficult to obtain in this case. In the next section we give an alternative characterization of orthogonal k -Circulant which helps in obtaining some insight into the nature of the possible solutions.

6.5 CHARACTERIZATION OF ORTHOGONAL k -CIRCULANTS

From Appendix B we have

$$H_k = F_k^{-1} \text{diag}(\lambda_0, \lambda_1, \dots, \lambda_{n-1}) F_k = F_k^{-1} \Lambda F_k \quad (6.14)$$

and $\Gamma_k = FD$ where $D = \text{diag}(1 \ p \ p^2 \ \dots \ p^{n-1})$

Using these in

$$H_k H_k^* = I_n \quad (6.15)$$

we obtain

$$F^* \Lambda F_k^{-1} \Lambda F_k = I_r$$

$$\text{or } D F^* \Lambda^{-1} D^{-1} D^{-1} D F^{-1} \Lambda FD = I_n$$

where $\Lambda = \text{diag}(\lambda_0 \ \lambda_1 \ \dots \ \lambda_{n-1})$

Noting that $F = \Gamma^{-1}$ we obtain

$$F \Lambda F (DD^*)^{-1} F^{-1} \Lambda F = D^{*-1} D^{-1} \quad (6.16)$$

$$\text{or } \Lambda^* F (DD^*)^{-1} F^{-1} \Lambda = F (DD^*)^{-1} F^*$$

$$\text{Let } X = F (DD^*)^{-1} F^{-1} \quad (6.17)$$

$$\text{Then } \Lambda^* X \Lambda = I \quad (6.18)$$

The problem of determining orthogonal k Circulant thus reduces to the determination of Λ such that (6.18) is satisfied for the given k . If none of the elements of γ are zero due to the diagonal nature of Λ it is easy to see that λ_1 's must satisfy the condition

$$\begin{bmatrix} \lambda_0 \\ \lambda_1 \\ \vdots \\ \lambda_{n-1} \end{bmatrix} [\lambda_0 \ \lambda_1 \ \dots \ \lambda_{n-1}] = E \text{ (matrix of all ones)} \quad (6.19)$$

To see this let $n = 2$ Then will be of the form

$$K = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix}$$

$$\text{then } \Lambda K \Lambda = \begin{bmatrix} \lambda_1^2 x_{11} & \lambda_1 \lambda_2 x_{12} \\ \lambda_2 \lambda_1 x_{21} & \lambda_2^2 x_{22} \end{bmatrix}$$

In order to satisfy (6.18) we must have

$$\lambda_1^2 = 1 \quad \lambda_1 \lambda_2 = 1 \quad \text{and} \quad \lambda_2^2 = 1$$

The procedure for obtaining an orthogonal k -Circulant is as follows. For the given value of k (6.18) must be solved to obtain λ_1 's which in turn produce H_k via (6.14). As an illustration we consider the case when $k = 1$.

When $k = 1$ $(DD^T)^{-1}$ becomes an identity matrix. Hence

$$X = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Using (6.18) we see that λ_1 's for an orthogonal circulant satisfy

$$\lambda_1^2 = 1 \quad \lambda_2 \lambda_n = 1 \quad \lambda_3 \lambda_{n-1} = 1 \quad \lambda_n \lambda_2 = 1$$

For a real matrix H_1 we have

$$\lambda_1 = \lambda_{n-1+2}^* \quad 1 = 2 \ 3 \quad n \quad \text{with} \quad \lambda_{n+1} = \lambda_1$$

Therefore

$$|\lambda_1|^2 = 1 \quad 1 = 1 \ 2 \quad \forall \quad \text{where} \quad 1 = \begin{cases} \frac{n}{2} + 1 & \text{if } n \text{ is even} \\ \frac{n+1}{2} & \text{if } n \text{ is odd} \end{cases}$$

Different choices for $\lambda_1, \lambda_2, \dots, \lambda_n$ decide different orthogonal real circulants. Thus we have infinite number of orthogonal Circulants. The λ_1 's are given by

$$\lambda_1 = \cos\theta_1 + j \sin\theta_1 \quad 0 \leq \theta_1 < 2\pi$$

For instance when $n = 4$ we can have

$$\lambda_1 = 1 \quad \lambda_3 = 1 \quad \lambda_2 = \cos\theta + j\sin\theta$$

Then the first row of the desired k -Circulant is given by

$$h_1^T = (1 + \cos\theta \quad \sin\theta \quad 1 - \cos\theta \quad \sin\theta)$$

6.6 LS SOLUTION USING k -CIRCULANT COMPLETION

From the analysis in the previous sections we conclude that for most cases of practical interest LS solution to the deconvolution problem cannot be obtained via k -Circulant completions as given by (6.1). However, although the solution given by (6.1) is not the LS solution it may be very close to the LS solution. In this section we obtain an alternate expression for the LS solution using k -Circulant completions.

Consider the received vector r given by

$$r = Hx + w \quad (6.20)$$

We first obtain the LS solution using heuristic arguments. If w were to be a zero vector and H_k^{-1} exists for some k , we can obtain exact solution to (6.20) as

$$x = x_1 = H_{k2}^{-1} r \quad \text{and} \quad x_2 = H_{k3}^{-1} r = 0 \quad (6.21)$$

where

$$H_k = [H \quad H_{k1}] \quad H_k^{-1} = \begin{bmatrix} H_{k2}^{-1} \\ 0 \end{bmatrix} \quad x_e = \begin{bmatrix} x \\ 0 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

On the otherhand when w is not a zero vector we have in general $x \neq x_1$ and $x_2 \neq 0$. Now consider x_2 given by $x_2 = H_{k3}^{-1} r = H_{k3}^{-1} (Hx + w)$. But $H_{k3}^{-1} H = 0_{n-m \times m}$. Therefore

$$H_{k3}^{-1} r = H_{k3}^{-1} w \quad (6.22)$$

The minimum norm least square solution (6.22) is given by

$$\hat{w} = H_{k3} (H_{k3}^{-1} H_{k3}^{-1})^{-1} H_{k3}^{-1} r \quad (6.23)$$

Thus we have estimated the noise vector w in the best possible manner under the given conditions. We can use this estimate to reduce the effect of noise in (6.20). Substituting for r using (6.20) in (6.21) for x_1 we obtain

$$x_1 = H_{k2}^{-1} r = x + H_{k2}^{-1} w$$

Using (6 23) for \underline{w} we get

$$H_{k2} x = x + H_{k2} H_{k3} (H_{k3} H_{k3}^*)^{-1} H_{k3} x$$

$$\text{or } x = H_{k2} [I - H_{k2} (H_{k3} H_{k3}^*)^{-1} H_{k3}] x$$

We now claim that no better result can be achieved under the given conditions. A formal proof of this claim is given below.

$$\text{Theorem 6.4} \quad P^+ = H_{k2} [I - H_{k2} (H_{k3} H_{k3}^*)^{-1} H_{k3}] \quad (6.24)$$

Proof We show that (6.24) satisfies all the four properties of the MP-inverse given by (6.10). Using (6.11) where instead of A_s we have H_k and H_1^{-1} replace A_s^{-1} with $H_{k1} = A_1$, $H_{k2} = A_{s2}$ and $H_{k3} = A_{s3}$ it is easy to show that representation of P^+ given by (6.24) satisfies the following conditions

$$HH^+H = H$$

$$H^+HH^+ = H^+$$

$$\text{and } (H^+H)^+ = H^+H$$

We have to finally show that $(HH^+)^+ = HH^+$. For this we make use of (6.11e) which gives after appropriate substitutions

$$HH_{k2} + H_{k1}H_{k3} = I_n$$

$$\text{Therefore } HH^+ = HH_{k2} [I - H_{k2}^* (H_{k3} H_{k3}^*)^{-1} H_{k3}]$$

$$= [I - H_{k1}H_{k3}] [I - H_{k2} (H_{k3} H_{k3}^*)^{-1} H_{k3}]$$

$$\begin{aligned}
&= [I - H_{k3} (H_{k3} H_{k3})^{-1} H_{k3}] - H_{11} H_{k3} + H_{11} H_{k3} \\
&= [I - H_{k3} (H_{k3} H_{k3})^{-1} H_{k3}]
\end{aligned}$$

which is a symmetric matrix

$$\text{therefore } (HH^T) = HH^T$$

But such a matrix is unique [35]

*4

In the foregoing discussion we used heuristic arguments for obtaining (6.24). Now we will show that (6.24) can be derived in a more formal manner. For this purpose we write (6.20) using k-Circulant completion of H and extended vector x_e as

$$\underline{r} = H_k x_e + w \quad (6.25)$$

Note that \underline{x}_e is an extended version of x obtained by padding x with zeros. However, while seeking solution to (6.25) via H_k^{-1} we ignored this constraint on \underline{x}_e . Therefore the solution naturally differs from x_{LS} except in some special cases. This means that in order to obtain LS solution to (6.25) we must minimize

$$(\underline{r} - H_k x_e)^T (\underline{r} - H_k \underline{x}_e) \quad (6.26)$$

subject to the constraint

$$C x_e = 0 \quad (6.27)$$

where $C = [O_{n-m \times n-g} \quad I_g]$

This minimization problem can be solved using Lagrange multiplier method. Let $\underline{\alpha} = (\alpha_1 \ \alpha_2 \ \dots \ \alpha_g)^T$ be the multipliers. Then we form

$$\Gamma(\underline{x} \ \underline{\alpha}) = (\underline{r} - H_k \underline{x}_e)^T (\underline{r} - H_k \underline{x}_e) + 2 \underline{\alpha}^T C \underline{x}$$

Differentiating $F(\underline{x} \ \underline{\alpha})$ with respect to \underline{x} and $\underline{\alpha}$ we obtain conditions for minimum as

$$H_k^* H_k \underline{x}_e - H_k^* \underline{r} - C^T \underline{\alpha} = 0$$

$$\text{and} \quad C \underline{x}_e = 0$$

As H_k is invertible we obtain

$$\underline{x}_e = H_k^{-1} \underline{r} - H_k^{-1} H_k^{*-1} C^T \underline{\alpha}$$

$$\text{and} \quad \underline{\alpha} = (C H_k^{-1} H_k^{*-1} C^T)^{-1} C H_k^{-1} \underline{r}$$

Existence of $(C H_k^{-1} H_k^{*-1} C^T)^{-1}$ follows by noting that

$$C H_k^{-1} = H_{k3} \quad \text{where} \quad H_k^{-1} = \begin{bmatrix} H_{k2} \\ H_{k3} \end{bmatrix}$$

and H_{k3} is a full rank matrix. Thus

$$\underline{x}_e = H_k^{-1} \underline{r} - H_k^{-1} H_k^{*-1} C^T (H_{k3} H_{k3}^*)^{-1} H_{k3} \underline{r}$$

Therefore the solution to the original problem is given by

$$\underline{x} = H_{k2} \underline{r} - H_{k2} H_{k3}^* (H_{k3} H_{k3}^*)^{-1} H_{k3} \underline{r}$$

Using Theorem 6.4 we see that x is the LS solution to the deconvolution problem

Using (6.24) we obtain an expression for the difference $\Delta H = H_{k2} - H^+$ given by

$$\Delta H = H_{k2} H_{k3}^* (H_{k3} H_{k3}^*)^{-1} + J \quad (6.28)$$

Note that ΔH is independent of H^+ . Therefore it can be computed by knowing H_k^{-1} .

If the elements of H are small the solutions obtained using H_{k2} and H^+ will not differ significantly. In the case of orthogonal k -Circulant completions of H it is easy to see that $\Delta H = 0_{m \times n}$ as $H_{k2} H_{k3}^* = 0_{m \times n-m}$.

In summary we have studied LS deconvolution problem in the framework of square completion of rectangular matrices. We have shown that in most of the practical situations LS solution cannot be obtained via direct inversion of k -Circulant completion of the channel convolution matrix.

Another method of obtaining LS solution is by using iterative schemes. In the next chapter we study steepest descent algorithm in the discrete Fourier transform domain which provides LS solution to the deconvolution problem.

CHAPTER 7

LEAST SQUARES DECONVOLUTION - ITERATIVE APPROACH

In the previous chapter we considered the possibility of applying computationally attractive k -Circulant completions of the channel matrix H for obtaining LS solution to the deconvolution problem which involves inversion of $H^T H$. Our analysis showed that in most of the practical cases the solution given by (6.1) using k -Circulant completions do not yield the LS solution to the deconvolution problem. An alternate expression for LS solution was also obtained in Section 6.6. Though the derivation of this expression makes use of k -Circulant completions the final expression is not computationally attractive compared to the direct inversion of $H^T H$.

In this chapter we study another method of obtaining LS solution to the deconvolution problem. It is based on the application of wellknown steepest descent algorithm (SDA) [89] which is an iterative technique in contrast to the noniterative or direct approach considered in Chapter 5.

Gersho [24] appears to have used fixed step SDA for the first time for the purpose of adaptive channel equalization in which the tap coefficients of the equalizer are adjusted continuously so that the mean square error between the actual and

estimated data symbols is minimized. He has shown that in the presence of noise the algorithm yields a suboptimal setting of the tap coefficients.

Walzman and Schwartz [21] [22] have studied fixed step SDA in the DFT domain for the purpose of a two-tap channel equalization. Using two examples they have observed that the DFT domain SDA converges faster than the sample domain SDA as exact knowledge of the eigenvalues are available in the case of DFT domain SDA. Their work has been extended by Picchi and Prati [64] who have included self-orthogonalizing feature into the DFT domain equalization scheme. Using simulation studies they have shown that the performance of self-orthogonalizing DFT domain equalization scheme is superior to that of the sample domain algorithm due to Godard [63].

The observations made by Walzman and Schwartz [21] and Picchi and Prati [64] are more carefully examined in this chapter. One reason for this is based on our understanding that DFT being an orthonormal transformation the iterative schemes in both the domains must have same convergence properties. But there may be computational advantages in going from sample domain to the DFT domain due to the availability of fast algorithms for computing DFT.

We devote a major portion of this chapter towards a better understanding of the role of DFT and k-Circulant completions in

reducing processing time and providing fast convergence. Our analysis clearly shows how DFT helps in reducing computations. Using the framework of k -Circular completions we show that advantage in terms of faster convergence rate can be achieved in the DFT domain because it provides in this case some examples good estimate of the convergence parameter. It is shown that the estimate obtained using k -Circular completion can also be used in the sample domain to obtain same convergence rate.

This chapter is organized in the following manner. A brief review of steepest descent algorithm is given in Section 7.1. The manner in which DFT helps in reducing computations is discussed in Section 7.2. In Section 7.3 we derive SDA in the DFT domain from three different viewpoints. Fast recovery scheme based on DFT domain SDA is given in Section 7.3.1. Convergence properties of SDA are discussed in Section 7.4. Method of increasing the convergence rate of the SD is discussed in Section 7.5. We show how Circular or k -Circular completions can be used to reduce number of iterations. This analysis provides explanation for the fast convergence behavior of the self-orthogonalizing DFT domain scheme studied by Picchi and Prati [64]. In Section 7.6 we show how estimates of step length in the case of fixed step SDA can be obtained using eigenvalues of the k -Circular completions. This explains the observation made by Milman and Schwartz [21] regarding

fast convergence of DFT domain scheme compared to sample domain scheme

1. REVIEW OF STEEPEST DESCENT ALGORITHM

In this section we discuss briefly certain general features of the wellknown Gauss step st descent algorithm [89]. Consider the quadratic objective function given by (2.8)

$$F(x) = x^T H^T H x - 2x^T H^T r - r^T r \quad (7.1)$$

The SD constructs a sequence of vectors starting with x_0 chosen arbitrarily given by

$$x_{1+1} = x_1 + \alpha_1 P^T (r - Hx_1) \quad \alpha_1 > 0 \quad (7.2)$$

where α_1 is determined such that $F(x_{1+1})$ is minimized. Substituting (7.2) in (7.1) and differentiating w.r. to α_1 it can be shown that α_1 is given by

$$\alpha_1 = \frac{e_1^T e_1}{e_1^T H^T H e_1} \quad e_1 = H^T (r - Hx_1) \quad (7.3)$$

Computation of α_1 at every iteration is the most time consuming part of SDA. Therefore in most of the practical applications α_1 is fixed at a value α chosen such that the optimum speed of convergence is achieved. It can be shown [89] that the iteration scheme (7.2) converges to the LS solution

$$x_{LS} = (P^T P)^{-1} H^T r$$

It may be interesting to note that the fixed step SDA ($\alpha = \alpha^*$ for all i) is essentially a generalized version of the Jacobi iteration [74] for solving the linear equation

$$H^T H x = H^T r$$

The iteration scheme in this case takes the form

$$x_{i+1} = (I_m - \alpha H^T H) x_i + \alpha H^T r \quad (7.4)$$

This iteration scheme can be shown to converge if $0 < \alpha < 2/\lambda_{\max}$. The optimum value of α is given by

$$\alpha = \frac{2}{\lambda_{\max} + \lambda_{\min}} \quad (7.5)$$

where λ_{\max} and λ_{\min} are the maximum and minimum eigenvalues of $H^T H$ respectively [90] (see page 24)

7.2 APPLICATION OF DFT TO STEEPEST DESCENT ALGORITHM

We now consider the possibility of using DFT to reduce computational load associated with SDA. Consider e_1 given by

$$e_1 = H^T (r - Hx_1)$$

Let H_1 be the Circulant completion of H and x_{1e} be the extended vector formed by padding x_1 with $n-m$ zeros. Then we can write

$$e_1 = H_1^T (r - H_1 x_{1e}) \quad (7.6)$$

where $e_1 = \begin{bmatrix} e_1 \\ - \\ e_{11} \end{bmatrix}$ and $e_{11} = H_1^T (r - H_1 x_{1e})$ $H_1 = [H \ H_{11}]$

From (7.6) it is clear that if we compute the right hand side of (7.6) \underline{e}_1 is obtained as the first n entries of \underline{e}_1 . Now let \underline{E}_1 denote the DFT of \underline{e}_1 . In terms of the DFT of \underline{x} \underline{E} and the first column of H_1 we can write (7.6) in the DFT domain as

$$\begin{aligned}\underline{z}_1 &= FH_1^T \Gamma (F\underline{x} - \Gamma H_1 F \Gamma \underline{y}_1) \\ &= F_1 (I - H_1) \underline{e}_1\end{aligned}\quad (7.7)$$

Therefore by knowing H_1 , P and \underline{z}_1 the right hand side of (7.7) is easily computed and \underline{e}_1 is then obtained as

$$\underline{e}_1 = PF^* \underline{E} \quad (7.8)$$

where $P = [I_m, O_{m \times r-m}]$

Now if \underline{E}_{1e} denotes the DFT of \underline{e}_{1e} where \underline{e}_{1e} is an extended vector obtained by padding \underline{z}_1 with $(n-n)$ zeros α_1 given by (7.3) can be written as

$$\alpha_1 = \underline{e}_1^T \underline{e}_1 / \underline{E}_{1e}^T (F_1^* H_1 \underline{E}_{1e}) \quad (7.9)$$

Thus by knowing the DFT of \underline{e}_{1e} computation of α_1 becomes relatively simple task as we already have the knowledge of H_1 .

It should be noted that the value of α_1 remains same both in the sample domain and the DFT domain. Accordingly the

convergence rate remains unaltered by the application of DFT. We will come back to this point later.

Further we wish to note that as k-DFT computation using FFT algorithm requires more computation compared to DFT k-Circulant completions other than $k = 1$ are not as useful in reducing computations per iteration compared to Circulant completion.

7.3 DERIVATION OF THE DFT DOMAIN STEEPEST DESCENT ALGORITHM

In this section we derive SDA in the DFT domain from three different viewpoints. The objective is to clearly bring out the role of DFT in SDA.

Approach 1 Using the extended vector \underline{x}_e and Circulant completion H_1 of H we can write (7.2) alternatively as

$$\underline{x}_{(1+1)e} = \underline{x}_{1e} + \alpha_1 P H_1^T (\underline{r} - H_1 \underline{x}_{1e})$$

where $P = \begin{bmatrix} P \\ 0_{n-n \times n} \end{bmatrix}$ (7.10)

Premultiplying the above equation by the DFT matrix we obtain

$$\underline{X}_{(1+1)e} = \underline{X}_{1e} + \alpha_1 F P H_1^T (\underline{r} - H_1 \underline{X}_{1e})$$

where $\underline{X}_{1e} = F \underline{x}_{1e}$. Since $F^H F = I_n$ we can modify the above expression as

$$\underline{X}_{(1+1)e} = \underline{X}_{1e} + \alpha_1 \underline{F} \underline{P} \underline{F}^* - \underline{H}_1^T \underline{F} (\underline{F} \underline{H}_1 \underline{F} \underline{F}^* \underline{X}_{1e}) \quad (7.11)$$

Now $\underline{F} \underline{H}_1 \underline{F}^*$ and $\underline{H}_1^T \underline{F}^*$ are diagonal matrices with DFT of the first column of \underline{H}_1 and its complex conjugate as diagonal elements respectively. Therefore we can write (7.11) as

$$\underline{X}_{(1+1)e} = \underline{X}_{1e} + \alpha_1 \underline{F} \underline{P} \underline{F}^* - \underline{H}_1^T \underline{F} (\underline{F} \underline{H}_1 \underline{F}^* \underline{X}_{1e}) \quad (7.12)$$

Moreover we can write (7.12) also as

$$\underline{X}_{(1+1)e} = \underline{X}_{1e} + \alpha_1 \underline{L}_1 \quad (7.13)$$

where

$$\underline{X}_{1e} = \underline{F} \underline{P} \underline{F}^* \underline{X}_{1e} \quad (7.14)$$

$$\underline{L}_1 = \underline{H}_1 (\underline{P} - \underline{H}_1 \underline{F} \underline{F}^* \underline{H}_1^T \underline{X}_{1e})$$

α_1 can be expressed as

$$\alpha_1 = \underline{F}_{1e}^T \underline{L}_{1e} / \underline{F}_{1e}^T (\underline{H}_1^* \underline{H}_1 \underline{F}_{1e}) \quad (7.15)$$

The pair (7.13) and (7.15) represents DFT domain relation schemes corresponding to (7.2) and (7.4) respectively.

Approach 2 In Approach 1 we derived DFT domain SDA starting from sample domain SDA. In the present approach we formulate the deconvolution problem as

$$\min_{\underline{z} \in R^n} (\underline{r} - H_1 \underline{z})^T (\underline{r} - H_1 \underline{z}) \quad (7.16)$$

subject to $C\underline{z} = 0$

$$\text{where } C = [O_{n \times m} \quad I_{n-m}] \quad (7.17)$$

The minimization problem given by (7.16) involves symmetric Circulant positive semidefinite matrix in general. It may be noted that even though $H^T H$ is positive definite $H_1^T H_1$ need not always be so. A simple illustrative example is given below.

Consider

$$H = \begin{bmatrix} 1 & 0 \\ -1 & 1 \\ 0 & -1 \end{bmatrix} \quad \text{clearly } H^T H = \begin{bmatrix} 2 & 1 \\ -1 & 2 \end{bmatrix} \text{ is positive definite}$$

whereas the Circulant completion of H

$$H_1 = \begin{bmatrix} 1 & 0 & -1 \\ -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix} \quad \text{gives } H_1^T H_1 = \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}$$

which has 0 as an eigenvalue. Consequently $H_1^T H_1$ is positive semidefinite.

It may be further noted that whereas minimization of $F(\underline{x})$ given by (7.1) is an unconstrained problem, (7.16) defines a constrained minimization problem. Such problems can be solved

using Rosen approach of projecting the gradient at every iteration on to the null space of C [89] [91]. Let G be a matrix such that $CG = O_{n-m \times n}$. From (7.17) it is easy to see that C can have the following form

$$G = \begin{bmatrix} I_m \\ - \\ O_{n-m \times m} \end{bmatrix}$$

Following the approach given in [89] (see pages 260-266), we can write down the iteration scheme for (7.16) as

$$z_{i+1} = z_i + \alpha_i CG^T H_1^T (r - H_1 z_i) \quad (7.18)$$

where $\alpha_i = \min_{\alpha} F(z_{i+1})$. But $CG^T = P$. We can now use DFT and write (7.18) as done in the case of Approach 1 in the DFT domain. Thus we obtain

$$z_{i+1} = z_i + \alpha_i FP^{-*} (H_1^T r - H_1^* H_1 z_i) \quad (7.19)$$

Since z_i plays the same role as x_i in (7.13) we see that (7.13) and (7.18) are identical.

Approach 3 We now consider the approach taken by Valzman and Schwartz [21] and Picchi and Prati [64] for deriving DFT domain SDA. Using F and the Circulant completion of F we can rewrite the minimization problem given by (7.16) as

$$\min_{y \in \mathbb{R}^n} (Fr - FH_1 y)^* F^{-*} F^{-*} (Fr - FH_1 y)$$

writing $FH_1\underline{y} = FH_1F^*F\underline{y}$ and noting that $F^*F = I_n$ we obtain

$$\min_{\underline{y} \in \mathbb{C}^n} (\underline{P} - F_1 \underline{y})^{*T} (R - H_1 \underline{y}) \quad (7.20)$$

where by our earlier convention $F = F^*$, $F_1 = F_1^*$ and $\underline{y} = F\underline{y}$. Now in terms of $\underline{y} \in \mathbb{C}^n$ we can formulate the minimization problem given by (7.16) equivalently as

$$\min_{\underline{y} \in \mathbb{C}^n} (\underline{P} - H_1 \underline{y})^{*T} (\underline{R} - F_1 \underline{y}) \quad (7.21)$$

subject to 1) $F\underline{y} \in \mathbb{R}^n$ real vector

and 2) $C F \underline{y} = 0$

where C is same as in (7.16). Solution to the above problem can be obtained using Rosen's gradient projection method as described in [21]. Alternatively since (7.21) is derived using (7.16) which has (7.19) as the corresponding SDA in the DFT domain it follows that (7.21) also will have (7.19) as its corresponding SDA in the DFT domain.

At this stage we wish to note the following distinctive feature of the three approaches. Approach 1 deals with a unconstrained minimization problem. In Approach 2 we deal with a constrained minimization problem with one constraint. In the third approach we have to deal with two constraints. The form of SDA in the DFT domain however remains unchanged. Likewise

The convergence parameter α_1 also remains unchanged by the application of DFT. Thus we can conclude that DFT and the associated Circulant completion help only in reducing the computational complexity of the iterative scheme. These aspects do not become clearer in the SDA derived using Approach 3 alone because solution of the minimization problem given by (7.21) does not show explicitly its relationship with SDA in the sample domain.

7.3.1 Data Recovery Scheme based on SDA in the DFT Domain

A DR scheme based on DFT domain iteration scheme (7.13) is schematically shown in Figure 7.1. In this scheme the starting point is chosen as

$$\underline{X}_{oe} = \underline{R} \underline{H}_1$$

Note that in the absence of noise and when the Circulant completion of \underline{H}_1 is nonsingular \underline{X}_{oe} represents the DFT of the actual data vector padded with $n-m$ zeros. Therefore in such situations the iteration scheme will converge in one iteration. On the other hand when noise is present \underline{X}_{oe} provides a good starting point for the iteration scheme. The performance of this scheme will be same as that of DR scheme based on LS solution.

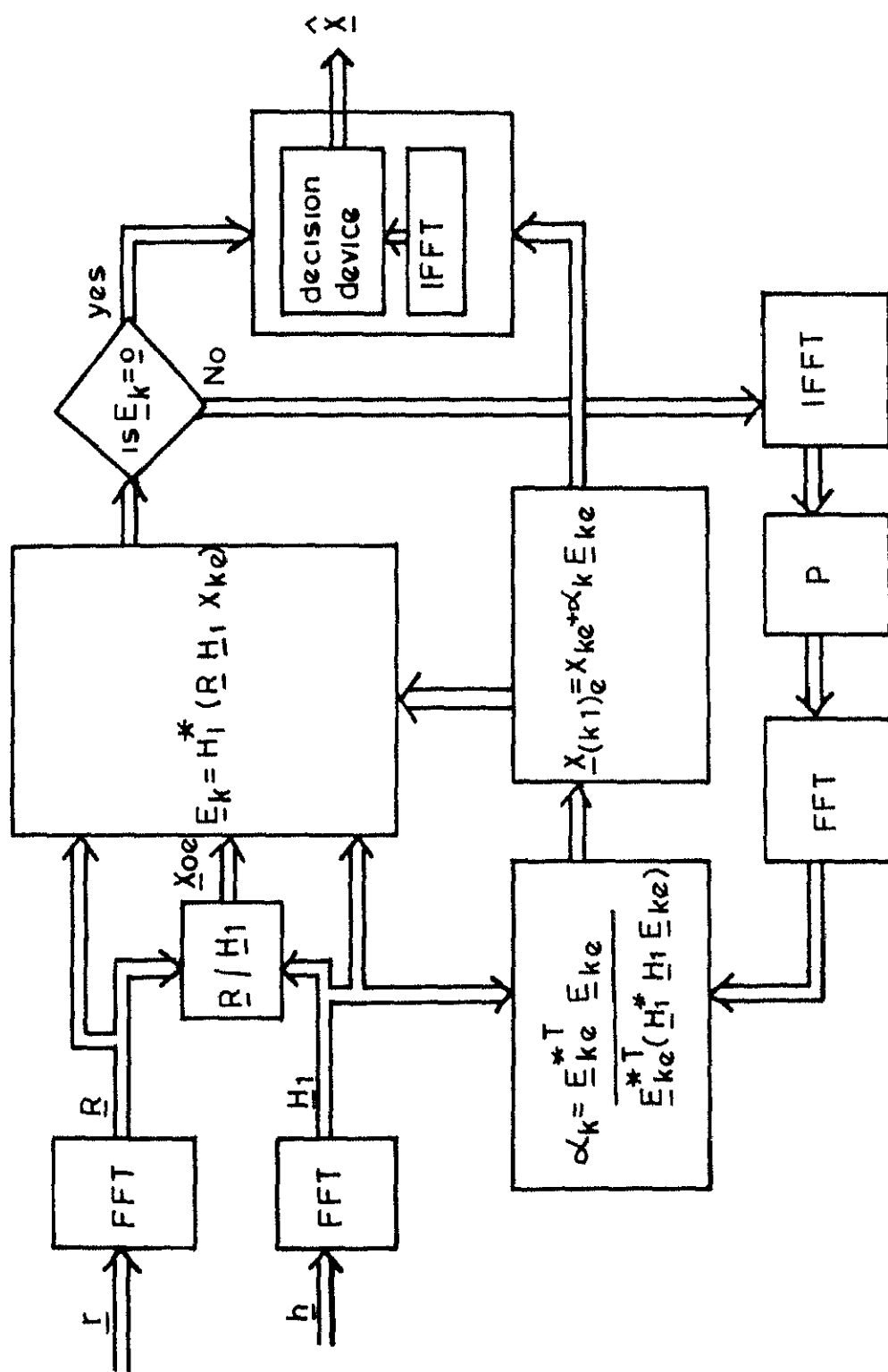


FIG 71 Data recovery scheme based on DFT domain steepest descent algorithm

7.4 RATE OF CONVERGENCE OF SDA

SDA generates an infinite sequence of \underline{x}_1 's whose limit is the solution to the optimization problem. Due to severe constraints on processing time in a communication situation, large number of iterations will not be a desirable feature. Therefore it becomes important to know the speed of convergence of SDA. In this connection we have the following theorem [89].

Theorem 7.1 Suppose that $\tilde{\underline{x}}$ is the solution to minimization of $F(\underline{x})$ given by (7.1). Let c_n be the condition number of $H^T H$. That is $c_n = \lambda_{\max} / \lambda_{\min}$. Then (see [89] pag. 146)

$$F(\underline{x}_{1+1}) - F(\tilde{\underline{x}}) \leq \left(\frac{c_n - 1}{c_n + 1} \right)^2 [F(\underline{x}_1) - F(\tilde{\underline{x}})]$$

$$\|\underline{x}_{1+1} - \tilde{\underline{x}}\| < c_n^{1/2} \left(\frac{c_n - 1}{c_n + 1} \right)^2 \|\underline{x}_1 - \tilde{\underline{x}}\|$$

From Theorem 7.1 it is clear that SDA has linear rate of convergence. The rate of convergence depends on the condition number of the channel autocorrelation matrix. If $c_n = 1$ the iteration scheme converges in one iteration. But as c_n increases, the convergence rate decreases and hence large number of iterations will be needed before meaningful estimate of the solution vector is obtained. Thus SDA is not an efficient method when the condition number of the channel is large. One method for accelerating the convergence rate of SDA is by

altering the eigenvalues of the channel autocorrelation matrix so that condition number is near to unity. In the next section we consider this aspect.

7.5 FAST CONVERGING STEEPEST DESCENT ALGORITHM

Consider the iteration scheme given by (7.2). The scheme converges to the exact solution when

$$H^T(r - Hx_1) = 0 \quad (7.22)$$

for some α . This means that in (7.2) we can substitute αQ where Q is a nonsingular matrix and obtain another iteration scheme

$$x_{i+1} = x_i + \alpha Q H^T (r - Hx_i) \quad (7.23)$$

which will also give the same solution as (7.2) provided α , Q and H satisfy the necessary convergent conditions. We can show that the iteration scheme (7.23) converges if

$$\|I_m - \alpha Q H^T H\| < 1$$

or $0 < \alpha < 2/\lambda_{\max}$ where λ_{\max} is the maximum eigenvalue of $Q H^T H$.

If $Q = (H^T H)^{-1}$ it is easy to see that the iteration scheme (7.23) converges to the solution in one iteration if we choose $\alpha = 2/(\lambda_{\max} + \lambda_{\min})$ where λ_{\min} is the minimum eigenvalue of $Q H^T H$ (when $Q = (H^T H)^{-1}$, $\lambda_{\min} = 1$).

It is clear from the above discussion that different choices of Q will lead to different iteration schemes. For $Q = I_m$ we obtain fixed step SDA. Since computation of the best choice for Q which is $(H^T H)^{-1}$ itself is uneconomical we must choose Q such that it is a good estimator of $(H^T H)^{-1}$ or the inverse of the channel autocorrelation matrix. Such choices are difficult to obtain in the sample domain.

Now let us look at the DFT domain representation of (7.23) given by

$$\underline{z}_{(1+1)e} = \underline{z}_{1e} + \alpha F Q_1 F^* \underline{E}_{1e} \quad (7.24)$$

where as before $\underline{y}_{1e} = F \underline{x}_{1e}$ and \underline{E}_{1e} is given by (7.14). Q_1 is now an $n \times n$ matrix. From (7.24) we observe that if $F Q_1 F^*$ is a diagonal matrix, additional computation needed for multiplying $F Q_1 F^*$ and \underline{E}_{1e} is just n multiplications. Therefore in order to make $F Q_1 F^*$ diagonal Q_1 must be a circulant matrix. Since the best choice for Q is $(H^T H)^{-1}$ it seems natural to choose for Q_1 $(H_1^T H_1)^{-1}$ whenever H_1 is nonsingular. This feature has been exploited by Picchi and Prati [64] in their fast converging self orthogonalizing DFT domain equalization scheme in the context of channel equalization.

So far in this section we discussed the application of Circulant completion for obtaining good estimates of the convergence parameter. However it may not be always possible

to obtain good estimate of the convergence parameter with Circulant completion. In such cases k -Circulant completions can be used. For this purpose let $Q_k = (H_k^* H_k)^{-1}$. We assume that H_k is invertible. When $|k| = 1$ it can be shown that Q_k is also a k -Circulant. Therefore making use of k -DFT (7.24) can be written using k -Circulant completion of H as

$$x_{(1+1)ke} = \underline{x}_{1ke} + \alpha F_k Q_k F_k^{-1} \underline{e}_{1e}$$

where $\underline{x}_{1ke} = F_k x_{1k}$

and $\underline{e}_{1ke} = F_k^* F_k^{-1} \underline{e}_{1k}$ $\underline{e}_{1k} = H_k^* (P_k - H_k \underline{x}_{1ke})$

Since Q_k is a k -Circulant matrix $F_k Q_k F_k^{-1}$ will be a diagonal matrix. It may be noted that use of k -Circulant completion requires computation of k -DFT which can be computed using FFT algorithm as described in Chapter 5. The efficiency of the iteration scheme depends on the value of k used. A direct analysis of the eigenvalue behaviour of Q_k and consequently its effect on α is difficult as it depends upon the channel given. This aspect needs more investigation.

7.6 FIXED STEP STEEPEST DESCENT ALGORITHM

The DFT domain version of fixed step SDA follows from our discussion in Section 7.3. In this case we can write (7.13) as

$$x_{(1+1)e} = \underline{x}_{1e} + \alpha \underline{e}_{1e} \quad (7.25)$$

The best choice of α is given by (7.5). As computation of λ_{\max} and λ_{\min} is a difficult task, estimates of the eigenvalues of the channel autocorrelation matrix can be used. An alternative method would be to choose the eigenvalues of $H_1^T H_1$ or $H_k^T H_k$ if they are good estimates or the eigenvalues of $\mathbf{H}^T \mathbf{H}$. One reason for such a choice is that they can be computed easily using DFT. Walzman and Schwartz [1] have used eigenvalues of $H_1^T H_1$ in the iterative scheme. However, they do not seem to have noted that α obtained using $H_1^T H_1$ can also be used in the sample domain.

Let $\lambda_{\max k}$ and $\lambda_{\min k}$ be maximum and minimum eigenvalues of $\mathbf{H}_k^T \mathbf{H}_k$ and let

$$\alpha^{(k)} = \frac{2}{\lambda_{\max k} + \lambda_{\min k}} \quad (7.26)$$

If $\alpha^{(k)}$ is close to α the degradation in the convergence rate of the iterative scheme will not be appreciable. Towards this end we prove the following theorem.

Theorem 7.2 Let λ_{\max} and λ_{\min} be the maximum and minimum eigenvalues of $\mathbf{H}^T \mathbf{H}$. Then $\lambda_{\max s} \leq \lambda_{\max}$

$$\lambda_{\min s} \leq \lambda_{\min} \quad (7.27)$$

where $\lambda_{\max s}$ and $\lambda_{\min s}$ represent maximum and minimum eigenvalues of $\mathbf{H}_s^T \mathbf{H}_s$ where \mathbf{H}_s is a nonsingular square completion of \mathbf{H} .

Proof We first prove that $||H^T H|| \leq ||H_s^T H_s||$

where $|| \cdot ||$ is the spectral norm of a matrix Let $P = \begin{bmatrix} I_n & 0 \\ 0 & 0 \end{bmatrix}$

be an $n \times n$ matrix Using P we can write

$$||H^T H|| = ||PH_s^T H_s P||$$

$$\text{for } ||PH_s^T H_s P|| = ||\begin{bmatrix} H^T H & 0 \\ 0 & 0 \end{bmatrix}|| = ||H^T H||$$

$$\text{Therefore } ||H^T H|| < ||P||^2 ||H_s^T H_s||$$

A P is a projection operator it has eigenvalue either 0 or 1

$$\text{Therefore } ||H^T H|| < ||H_s^T H_s||$$

If $\lambda_{\max s}$ and λ_{\max} are the maximum eigenvalues of $H_s^T H_s$ and $H^T H$ respectively we obtain

$$\lambda_{\max} \leq \lambda_{\max s}$$

That proves the first inequality in (7.27)

$$\text{We now show that } ||(H^T H)^{-1}|| \leq ||(H_s^T H_s)^{-1}||$$

$$\text{we have } H_s^T H_s = \begin{bmatrix} H^T H_c & H^T H_c \\ H^T H_c & H^T H_c \end{bmatrix}$$

Note that $H_s = \begin{bmatrix} H & H_c \end{bmatrix}$ Consider the matrix G

$$G = \begin{bmatrix} I & 0 \\ -(H_c^T H)(H^T H)^{-1} & I \end{bmatrix}$$

Then $||G^{-1}|| = 1$ Therefore

$$||(\mathbf{H}_s^T \mathbf{H}_s)^{-1}|| = ||(\mathbf{H}_s^T \mathbf{H}_s)^{-1}|| ||G^{-1}||$$

Using the result $||AB|| \leq ||A|| ||B||$ we obtain

$$||(\mathbf{H}_s^T \mathbf{H}_s)^{-1}|| > ||(\mathbf{H}_s^T \mathbf{H}_s)^{-1} \mathbf{G}||$$

$$\text{Now } \mathbf{G} \mathbf{H}_s^T \mathbf{H}_s = \begin{bmatrix} \mathbf{H}^T \mathbf{H} & \mathbf{H}^T \mathbf{H}_c \\ 0 & \mathbf{H}_c^T \mathbf{P}_c - \mathbf{H}_c^T \mathbf{H} (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{H}_c \end{bmatrix} = ||(\mathbf{G} \mathbf{H}_s^T \mathbf{H}_s)^{-1}||$$

$$\text{Therefore } ||(\mathbf{G} \mathbf{H}_s^T \mathbf{H}_s)^{-1}|| \geq ||(\mathbf{H}^T \mathbf{H})^{-1}||$$

$$\text{Therefore } ||(\mathbf{H}^T \mathbf{H})^{-1}|| < ||(\mathbf{H}_s^T \mathbf{H}_s)^{-1}||$$

$$\text{Therefore } \lambda_{\min s} \leq \lambda_{\min}$$

where $\lambda_{\min s}$ and λ_{\min} are the minimum eigenvalues of $\mathbf{H}_s^T \mathbf{H}_s$ and $\mathbf{H}^T \mathbf{H}$ respectively

**

In particular for k-Circulant completions we have

$$\lambda_{\max k} \geq \lambda_{\max}$$

$$\text{and} \quad (7.28)$$

$$\lambda_{\min k} \leq \lambda_{\min}$$

In a given situation for fixing α it may be desirable to try out different k-Circulant completions and choose the one which provides maximum α . But this involves additional computations. The problem of maximizing α over all possible

k -Circulant completions for a given channel does not appear to admit mathematically tractable solutions. This aspect however needs further investigation.

In summary we have discussed the role of DFT and k -Circulant completions in reducing the computational complexity and increasing the convergence rate of SDA. We have showed how to obtain better estimates of the convergence parameter α using eigenvalues of k -Circulant completions of channel convolution matrix H . As there is a one-to-one correspondence between sample domain and DFT domain iteration schemes estimates of α obtained via k -Circulant completion can be used in the sample domain itself to provide faster convergence.

SDA has only linear convergence property and the rate of convergence depends upon the condition number of the channel autocorrelation matrix. In this connection self orthogonalizing scheme discussed in Section 7.5 could be useful.

Considering the drawbacks of SDA despite its several advantages it appears that when there is severe constraint on processing time a better method will have to be chosen. The method so chosen must converge fast and it must have a better rate of convergence. We wish to note that conjugate gradient algorithm (CGA) [89] meets these demands. However it is computationally unattractive because each iteration takes more

processing time compared to SDA. Kobayashi [92] has considered CGA for adaptive channel equalization. However, CGA does not appear to have received much attention in the area of adaptive channel equalization due to its computational complexity.

CHAPTER 8

PERFORMANCE OF DATA RECOVERY SCHEMES

In this chapter we give a comparative account of performance of some of the data recovery schemes studied in previous chapters. Performance is specified in terms of probability of error (P_e). Since the probability of error depends upon block length, channel parameters, and recovery schemes employed, noise statistics and the rate of transmission over the channel, except in very special cases, in general, it is very difficult to obtain closed form expressions for P_e . In this connection, it may be recalled that it is possible to obtain closed form expressions for P_e for DR schemes considered in Chapter 5. On the other hand, the schemes considered in Chapters 3 and 4 are non-linear in nature and hence are difficult to analyze. Under such situations, simulation studies help in evaluating the performance of the DR scheme employed. Throughout this chapter, we assume binary data transmission. However, the analysis can also be extended to multilevel data.

This chapter is organized in the following manner. As a prelude to the study of the performance of various DR schemes, in Section 8.1 we characterize good channels for BDT which introduce ISI and provide good performance subject to a given

criterion. There are two reasons for knowing which channels. Firstly, they provide a benchmark for comparing the performance of other channels. Secondly, one could use a prefilter at the transmitter so as to make the overall response of the channel and the prefilter to resemble closely a good channel. Channel examples used in the simulation study are given in Section 8.2. In Section 8.3 we discuss performance of BDT systems using performance curves for a wide range of block lengths and typical channel examples. In Section 8.4 we study the effect of increasing ISI among the members of a block which happens when the data rate over the channel is to be kept the same as source data rate.

8.1 GOOD CHANNELS FOR BLOCK DATA TRANSMISSION

8.1.1 Maximum Minimum Distance Channels

Let S_x be the set of all possible m -length data vectors. We know from our discussion in Section 2.4 that when r is received in the presence of white Gaussian noise the best estimate of the transmitted vector \underline{x} is given by a vector $\underline{x}_1 \in S_x$ such that

$$\|\underline{r} - H\underline{x}\| \leq \|\underline{r} - H\underline{x}_j\| \quad \text{for all } \underline{x}_j \in S_x \quad (8.1)$$

Let S_y denote another set defined as

$$S_y = \{\underline{y}_j \mid \underline{y}_j = n\underline{x}_j, \quad \underline{x}_j \in S_x\} \quad (8.2)$$

where v_j is the received vector corresponding to x_j when noise is absent. Let d_{1j} denote the distance between two distinct vectors x_1 and x_j . Then

$$d_{1j} = \|x_1 - x_j\| \quad 1 \neq j \quad (8.3)$$

Due to the presence of noise the deconvolution rule (8.1) need not result in x_1 which is the actual data vector transmitted. In order to minimize the occurrence of such situations it is desirable to have good separation between any pair of x_j 's in terms of the distance metric given by (8.3). Let the minimum distance be d_{\min}

$$d_{\min} = \min_{x_1 \neq x_j \in S_y} \|x_1 - x_j\| \quad (8.4)$$

Without loss of generality we will consider channel responses with unit length. A $g+1$ length vector \underline{f} is called maximum minimum distance (MMD) channel sample response if

$$d_{\min} = \min_{1 \neq j} d_{1j} \quad 1 \leq 1 \leq j \leq 2^m \quad (8.5)$$

subject to $\underline{f}^T \underline{f} = 1$ and $-1 < f_1 < 1$ $0 \leq 1 < g$ is maximum

The second constraint in the above formulation is necessary to avoid trivial solutions for example $f_1 = 1$ and $f_j = 0$ for $j \neq 1$ $0 \leq j < g$

d_{1j} in (8.3) can be written as

$$d_{1j}^2 = \|x_1 - x_j\|^2 = \|Hx_1 - Hx_j\|^2 = (x_1 \ x_j)^T H^T (x_1 - x_j)$$

Since convolution operation is commutative we can interchange the roles of x_1 and h and write

$$d_{1j}^2 = f^T (\lambda_1 - \lambda_j)^T (\lambda_1 - \lambda_j) f \quad (8.6)$$

where $f_1 = h_1$ $0 \leq i \leq q$

and λ_1 is the convolution matrix with h_1 in the first column. Let

$$\lambda_{1j} = X_1 - X_j \quad (8.7)$$

It then follows that the elements of Z_{1j} are binary variables (+2 0 -2). Moreover λ_{1j} is also a convolution matrix. We can now write d_{1j}^2 alternatively as

$$d_{1j}^2 = f^T Q_{1j} f \quad (8.8)$$

where Q_{1j} is a symmetric Toeplitz matrix. Q_{1j} is given by

$$Q_{1j} = Z_{1j}^T Z_{1j}$$

In terms of (8.8) the optimization problem (8.5) becomes

$$\begin{aligned} \max_{f \in R^{q+1}} \quad & \min_{\substack{1 \leq j \\ 1 \leq i \leq 2^m}} f^T Q_{1j} f \end{aligned} \quad (8.9)$$

$$\text{subject to } 1) \ f^T f = 1 \quad (8.10)$$

and $2) \ -1 < f_1 < 1 \quad 0 \leq i \leq q$

The following lemma provides maximum value achievable if we solve (8.9) subject to (8.10)

Lemma 8.1 In the case of binary alphabet the maximum d_{\min} achievable is at most 2

Proof We make use of the inequality

$$\lambda_{\min 1j} \underline{f}^T \underline{f} \leq \underline{f}^T Q_{1j} \underline{f} \leq \underline{f}^T \underline{f} \lambda_{\max 1j} \quad (8.11)$$

where $\lambda_{\min 1j}$ and $\lambda_{\max 1j}$ are the minimum and maximum eigenvalues of Q_{1j}

Consider two data vectors which differ in only one position. In this case Q_{1j} is a diagonal matrix with all diagonal entries equal to 4. Therefore

$$\lambda_{\min 1j} = \lambda_{\max 1j} = 4 \quad (8.12)$$

Using the condition $\underline{f}^T \underline{f} = 1$ in (8.11) we get

$$\lambda_{\min 1j} \leq \underline{f}^T Q_{1j} \underline{f} \leq \lambda_{\max 1j} \quad (8.13)$$

From (8.12) and (8.13) it is clear that we cannot find an \underline{f} for all $i \neq j$ such that

$$\underline{f}^T Q_{1j} \underline{f} > 4 \quad \text{for all } i \neq j \quad **$$

Using Lemma 8.1 we prove the following theorem

Theorem 8.1 Determination of \underline{f} which gives $d_{\min} = 2$ is equivalent to solving

$$\underline{f}^T Q_{ij} \underline{f} \geq 4 \quad \text{for all } i \neq j \text{ and } \lambda_{\min ij} < 4 \quad (14)$$

$$\underline{f}^T \underline{f} = 1$$

$$-1 < f_i < 1 \quad 0 \leq i \leq g$$

Proof If \underline{f} is chosen such that for all of those Q_{ij} with $\lambda_{\min ij} < 4$ we have

$$\underline{f}^T Q_{ij} \underline{f} \geq \quad \text{clearly for the res of the } Q_{ij} \text{ s}$$

$$\underline{f}^T Q_{ij} \underline{f} > 4 \text{ holds from (13)}$$

**

According to Theorem 8.1 the procedure for obtaining MMD channels is to first determine all those Q_{ij} s for which $\lambda_{\min ij} < 4$. Then we have to solve the set of quadratic inequalities given by (8.14). While good amount of literature is available on linear inequalities [93] very little seems to have been achieved in understanding of systems of quadratic inequalities. We will not solve this problem directly in this thesis. Instead we will consider another characterization of good channels called maximum distance channels (MD-channels). We shall see later that MD-channel characterization is helpful in obtaining MMD channels.

8.1.2 Maximum Distance Channels

We now consider those channels which maximize the sum of distinct distances between all possible pairs of elements in S_y . That is, if the pairs y_1, y_j and y_p, y_q give the same distance ($d_{1j} = d_{pq}$) only one of the pairs will be considered in the summation. In this case the problem may be formulated as

$$\max_{\underline{f} \in R^{g+1}} \sum_{Q_{1j} \text{ distinct}} \underline{f}^T Q_j \underline{f} \quad (8.15)$$

subject to (8.10)

put differently (8.15) can also be written as

$$\max_{\underline{f} \in R^{g+1}} \underline{f}^T Q \underline{f} \quad (8.16)$$

$$\text{subject to (8.10) where } Q = \sum_{Q_{1j} \text{ distinct}} Q_{1j} \quad (8.17)$$

In the presence of the second condition in (8.10) the problem given by (8.16) is difficult to analyze. We therefore solve the following simpler problem

$$\max_{\underline{f} \in R^{g+1}} \underline{f}^T Q \underline{f} \quad (8.18)$$

subject to $\underline{f}^T \underline{f} = 1$

A solution to problem (8.18) can be obtained by using (8.11) from which it follows that the eigenvector corresponding

to the maximum eigenvalues is the desired solution to (8 18). If this solution satisfies the second constraint in (8 10) then a solution to (8 16) is obtained provided the minimum distance achieved is 2 in the binary case. However the procedure outlined above is computationally unattractive for large block sizes. Moreover it has not been possible to sort in conditions under which solution to (8 18) is also a solution to (8 16). We feel that a more rigorous study may lead to interesting results. In the absence of such results we have followed the procedure outlined below for obtaining ID as well as MD channels.

8 1 3 Procedure for Obtaining WID and ID Channels

- 1 Given the block length m and number of ISI samples g form the Q matrix given by (8 17)
- 2 Determine the normalized eigenvectors and the corresponding eigenvalues of Q
- 3 The eigenvector corresponding to the maximum eigenvalue is a solution to (8 18)
- 4 Compute the minimum distance for each eigenvector satisfying the second constraint in (8 10)
- 5 The eigenvector giving maximum minimum distance is a solution to (8 14) if the maximum minimum distance is 2 in the binary case

It must be noted that we may not get solution to (3.14) as given by the last step in all the cases. Towards this end we conducted some numerical experiments to obtain channel responses using the above procedure for several values of n and g . Some of these results are presented in Table 8.1. In the last column of the table we have given the minimum distances achieved. In all the cases, we have obtained both MID as well as MD channels. Another interesting feature is that some of the channel responses are reminiscent of the partial response channels [94]. For instance in the case of two ISI terms the channel responses $(-707 \ 0 \ 707)$ and $(-707 \ 0 \ -707)$ are duobinary channel responses. The channels obtained for $g = 4 \ 5 \ 6$ also show similar features.

The foregoing observation is interesting because partial response signalling schemes were originally suggested in order to achieve better spectral shaping of the transmitted signal. As we do not have yet a formal definition of such schemes it is difficult to establish meaningful relation between spectral shaping and maximum distance separation. This aspect needs further investigation. At this point we wish to note that Folconer and Magee [47] also have shown that optimum desired impulse of the channel and prefilter combination of length two under minimum mean square criterion must be the duobinary impulse response.

Table 8 1 Some examples of ID and MMD Channels

m	g	Channel sampled response										C_{min}	Remarks
6	3	707	0 0	0 707								2	ID and ID
		- 707	0 0	0 707								2	D
6	4	707	0 0	707	0 0							2	ID nd D
		707	0 0	- 707	0 0							2	ID
		0 0	- 707	0 0	- 707							2	ID and D
		0 0	- 707	0 0	707							2	ID
6	5	-0 8165	0 0	0 40825	0 0	0 40825						2	D
		0 57735	0 0	0 57735	0 0	0 57735						1 633	D
		0 0	707	0 0	707	0 0						2	ID
6	6	-0 8165	0 0	0 408	0 0	0 408	0 0					2	ID
		0 577	0 0	0 577	0 0	0 577	0 0					1 633	D
7	7	-0 86603	0 0	0 2886	0 0	0 2886	0 0	0 2886				2	ID
		0 5	0 0	0 5	0 0	0 5	0 0	0 5				1 414	ID

IID channels with $d_{\min} = 2$ have another interesting property when used in conjunction with L-d convolution or Viterbi algorithm. For such channels there will be essentially no loss in SNR when ML-deconvolution method is adopted. This conclusion is based on the analysis given by Forney [38]. He has shown that in the case of maximum likelihood sequence estimation the symbol error probability is bounded by

$$K_L Q(d_{\min}/N_0) < P_e \leq K_U Q(d_{\min}/N_0) \quad (8.19)$$

where

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-y^2/2} dy \quad (8.20)$$

and K_L and K_U are constants independent of noise variance and the channel parameters. K_L is typically within an order of magnitude of K_U . We know that for one shot transmission (no ISI) the probability of error is given by

$$P_e(\text{one shot}) = Q(2||\underline{f}||/N_0) \quad (8.21)$$

Thus if $d_{\min} = 2||\underline{f}||$ the probability of error achieved in the presence of ISI using ML-deconvolution procedure will be approximately equal to $P_e(\text{one shot})$.

8.1.4 Computation of Minimum Distance of a Channel

A direct method of determining d_{\min} is to compute the distance between all possible pairs of \underline{y} 's belonging to set S_y .

given by (8.2). This is a difficult job even for moderate block lengths. We now show that NL-PDA d v 1 d 11 Chapter 3 can be modified for computing d_{\min} .

From our earlier discussion in this chapter it is clear that the problem of finding minimum distance can be formulated as

$$\min_{\substack{x_1, x_j \in S_x \\ 1 \neq j}} (x_1 - x_j)^T H^T H (x_1 - x_j)$$

$$\text{Let } z_{1j} = x_1 - x_j$$

Elements of z_{1j} are ternary variables taking values from the set $\{-2, 0, 2\}$. Let \underline{z} denote a vector whose elements are ternary variables. In terms of \underline{z} the above problem can be alternately formulated as

$$\min_{\underline{z}} \underline{z}^T H^T H \underline{z} \quad (8.22)$$

$$\text{subject to } \underline{z}^T \underline{z} > 0 \quad (8.23)$$

Note that the constraint $\underline{z}^T \underline{z} > 0$ is necessary to eliminate all zero solution for cases corresponding to $x_1 = x_j$.

The foregoing problem is a quadratic optimization problem in the presence of a constraint. In order to find the non-zero minimum value we adopt the procedure of Chapter 3 with the following difference. In the first step when z_1 is optimized for every given value of z_2, z_3, \dots, z_{g+1} choose only non zero

z_1 's. Rest of the recursion stage are carried out without any constraint. This follows from the fact that since z_1 is constrained to take only nonzero values the constraint (8.23) is satisfied and hence rest of the z_1 's can be chosen arbitrarily. We wish to note that in the context of CSDT Burkhart and Barbosa [95] have suggested a modified Viterbi algorithm for determining the minimum distance of a given channel.

8.2 CHANNELS USED IN THE STUDY

In order to compare the performance of different data recovery schemes it is necessary to consider channels which show both merits as well as demerits of various DR schemes. The channel examples considered are given in Table 8.2. In Table 8.2 the first two channels A1 and A2 are maximum distortion channels considered by Austin [13]. The next two

Table 8.2 Channels used in the simulations

Channel sampled response		d_{\min}
A1	$3^{-1/2}(1 \ 1 \ 1)$	1.633
A2	$5^{-1/2}(1 \ 1 \ 1 \ 1 \ 1)$	1.265
C1	$2^{-1/2}(-235 \ 667 \ 1 \ 667 \ -235)$	1.933
C2	$2^{-1/2}(235 \ 667 \ 1 \ 667 \ 235)$	1.188
P1	$(227 \ 460 \ 688 \ 460 \ 227)$	1.131
P2	$(407 \ 815 \ 407)$	1.630

examples C1 and C2 taken from [4] The first two examples P1 and P2 are taken from [7]

Maximum distortion channels as the name itself indicates exhibit maximum realizable distortion according to some distortion measure Austin [13] determined such channels by choosing the samples of the channel autocorrelation coefficients $a(-g)$ $a(0)$ $a(g)$ so as to maximize

$$\sum_{l=-g}^g |a(l)|$$

subject to $|a(0)|$ is unity

and
$$2 \quad A(w) = \sum_{l=-g}^g a(l) \exp -j l w$$

$$1 + 2 \sum_{l=1}^g a(l) \cos(lw) > 0$$

when $w \in (0, \pi)$

The second condition implies that the channel is realizable Austin has shown that the sampled autocorrelation function having a triangular envelope corresponds to maximum distortion channels in the sense defined above

8.3 SIMULATION STUDIES ON DATA RECOVERY SCHEMES

The following data recovery schemes are considered

- | | | |
|---|----------------------------------|-----|
| 1 | ML data recovery scheme for CSDT | MLC |
| 2 | ML data recovery scheme for BDT | MLB |

3	Decision feedback scheme for CSDT	DFC
4	Modified Austin's approach for BDT	MAB
5	Fourier expansion method with decision feedback	FEB
6	LS data recovery scheme for BDT	LSB
7	Circulant completion based scheme for BDT	CCB
8	Screw Circulant completion based scheme for BDT	SCB

Note that a three letter identifier is associated with each data recovery scheme. These identifiers have been used throughout in the text, tables and figures. We wish to note that LSB is based on Fourier expansion method without decision feedback. In all the simulations 10000 data symbols taking binary (+1 -1) values were used. The curves plotted are obtained using single simulation.

In Figures 8.1 - 8.9 we have shown the variation of the probability of error with SNR for the six channels given in Table 8.2. Performance curves in the case of ML-C are obtained using ML-RDA given in Section 3.2 with the following modification. Decision regarding a transmitted data is made after a fixed no. of recursions say ID. That is after the IDth stage we follow the path corresponding to the maximum value of the objective value achieved so far and choose the data value corresponding to first stage. After that we shift the ID stage results one stage back so that the second stage now becomes the

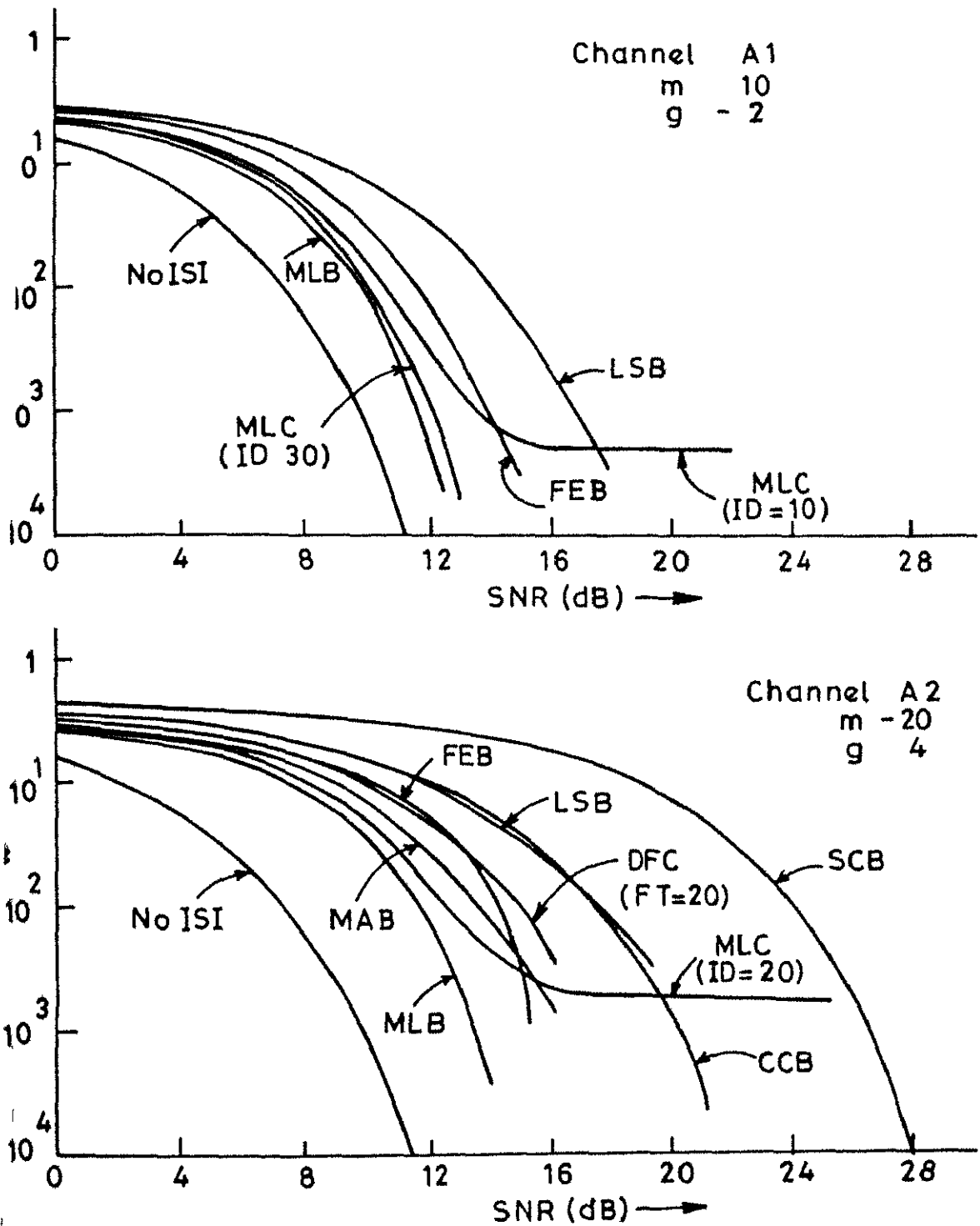


FIG 8 1 Comparison of performance of various DR scheme

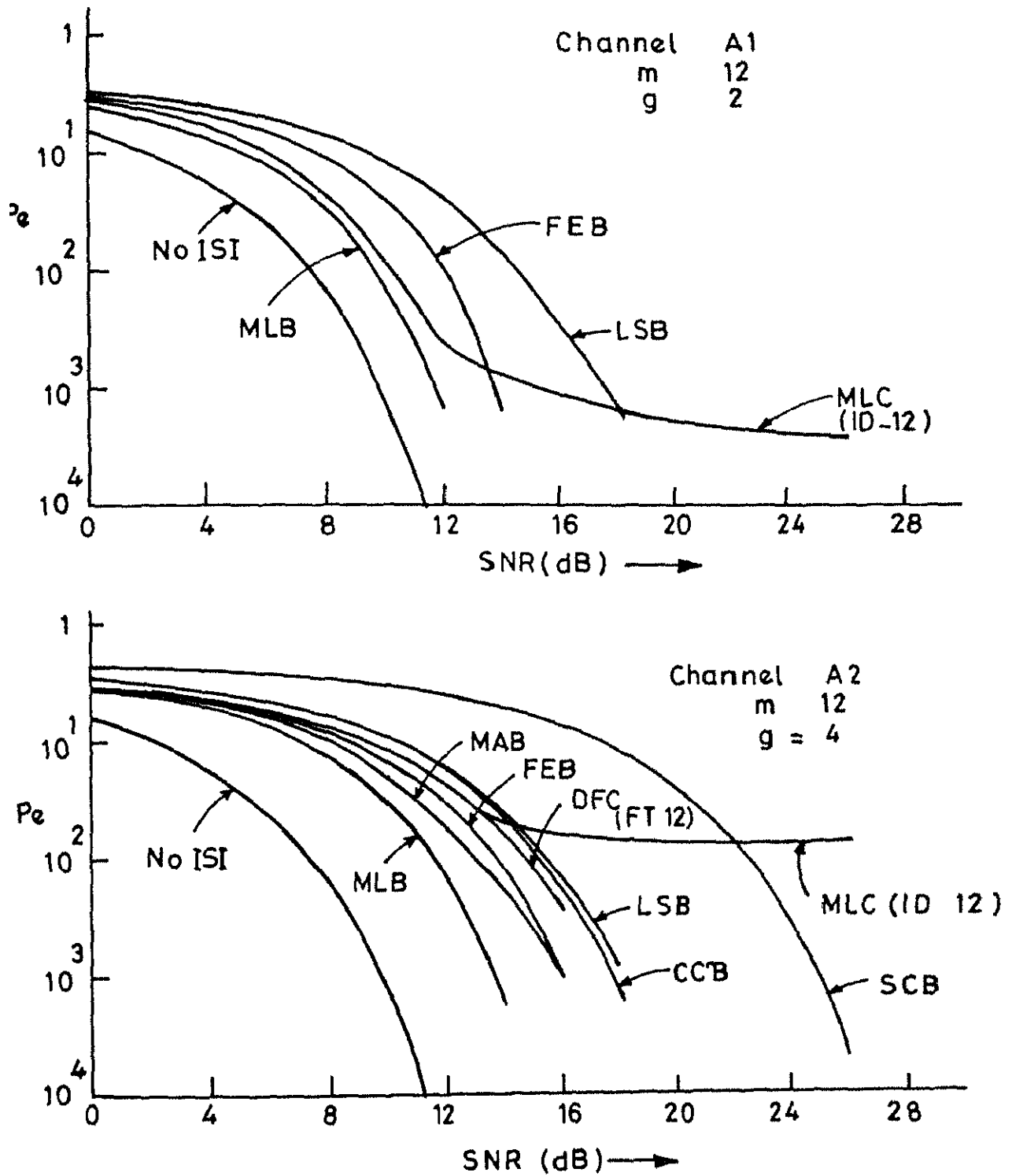


FIG 8 2 Comparison of performance of various DR schemes

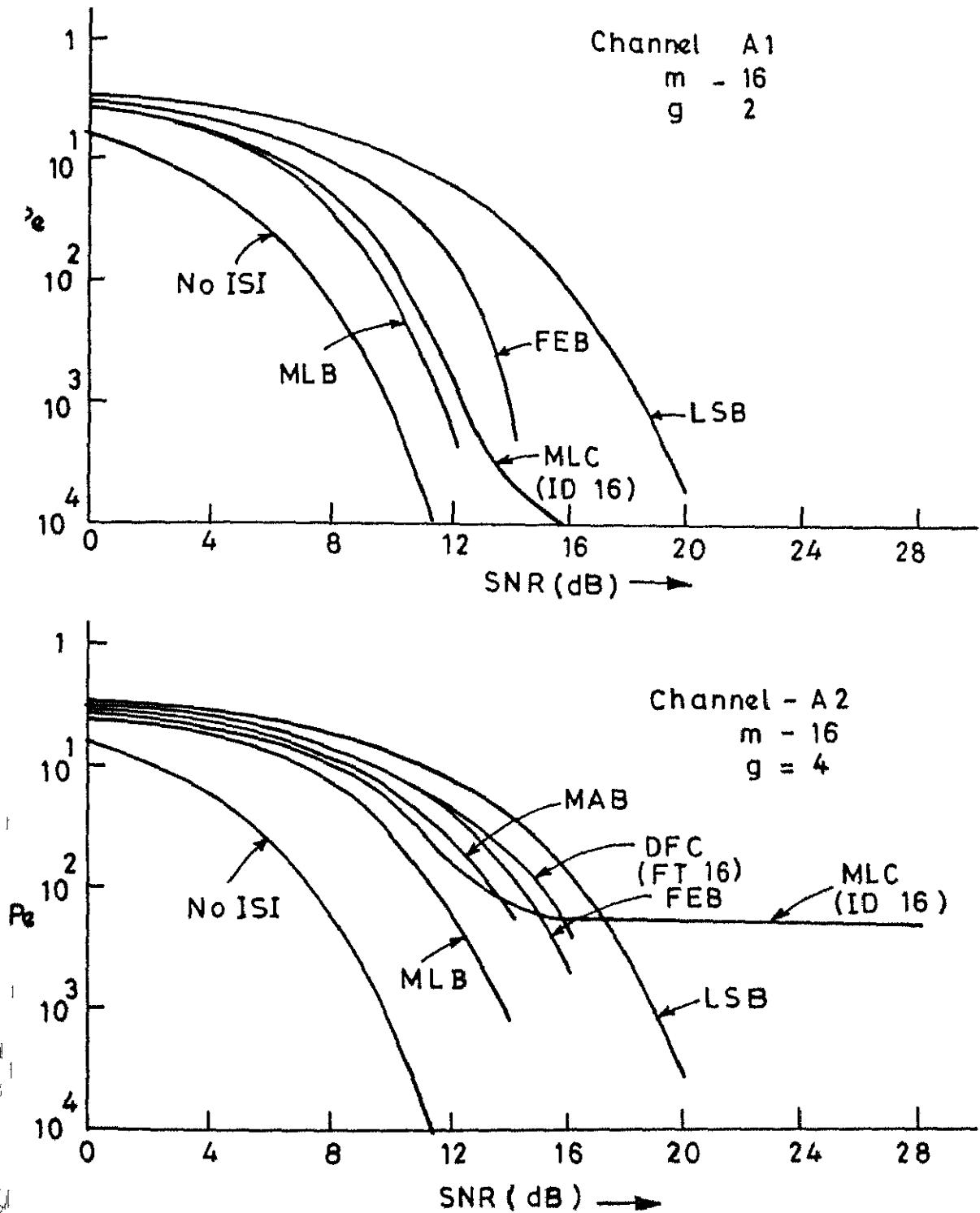
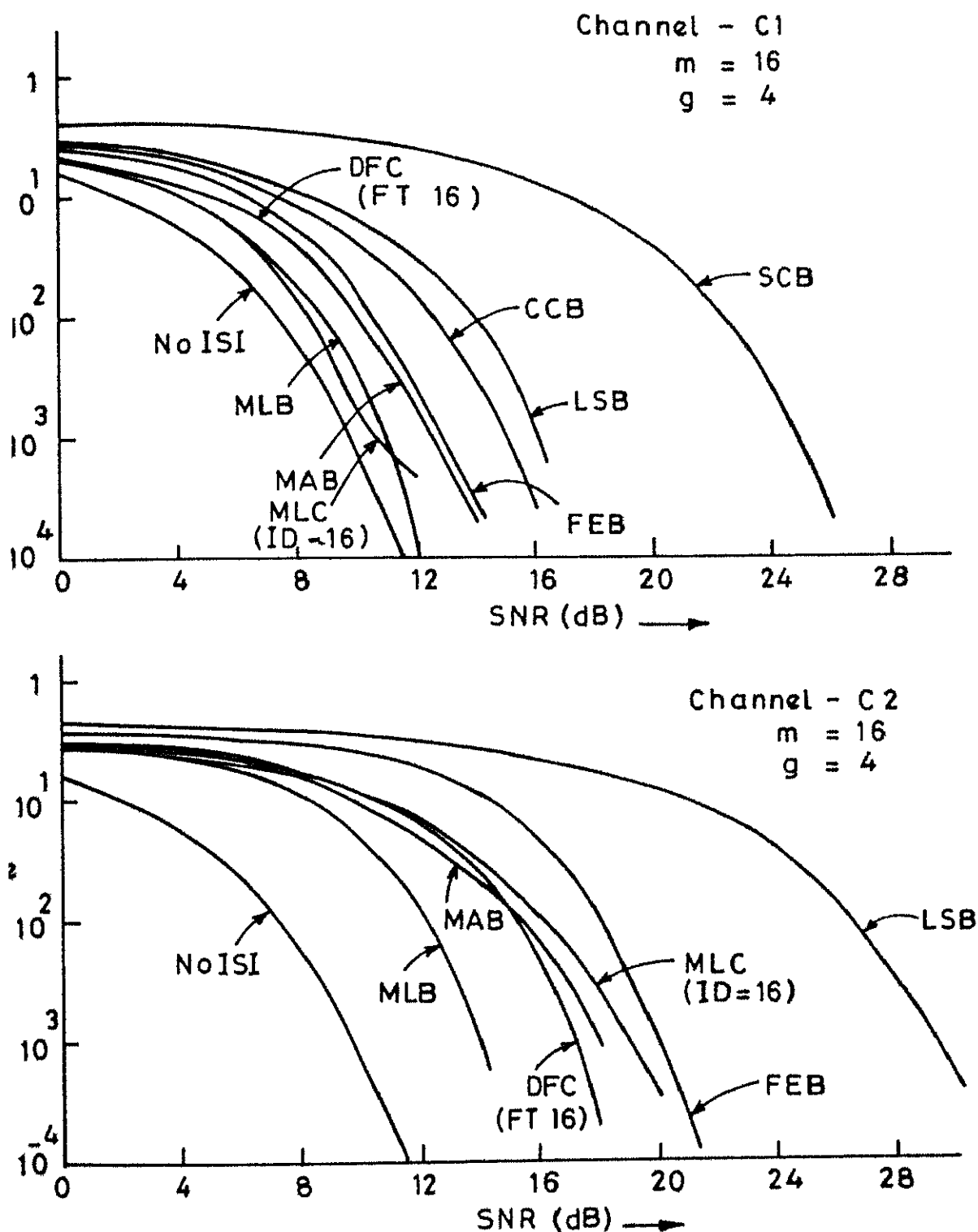


FIG 8 3 Comparison of performance of various DR schemes



IG 8 4 Comparison of performance of various DR schemes

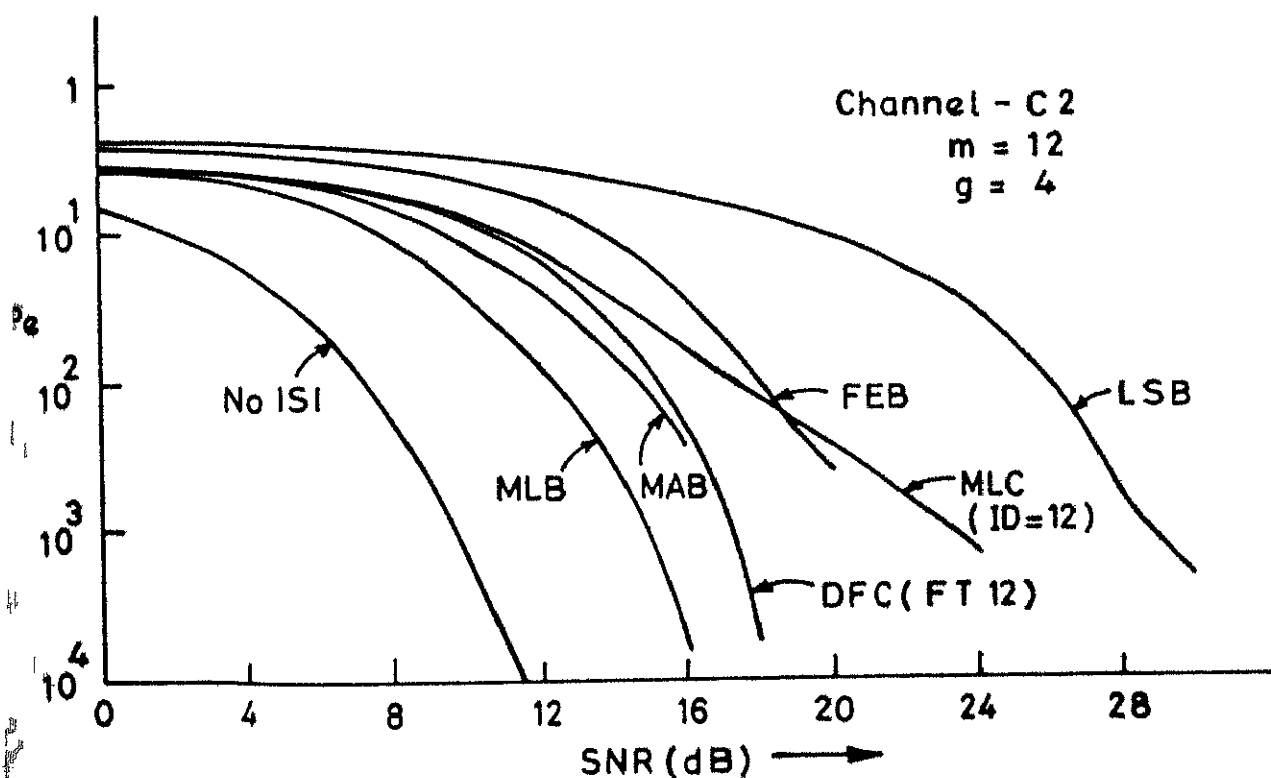
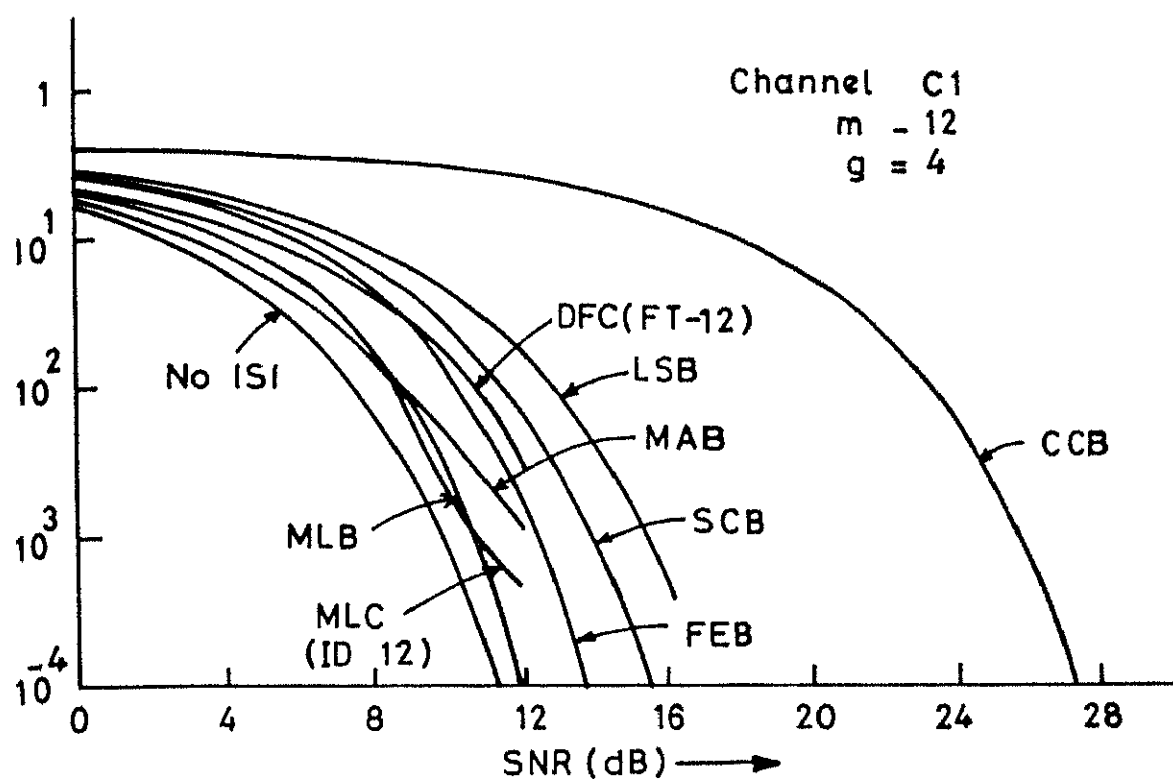


FIG 8 5 Comparison of performance of various DR schem

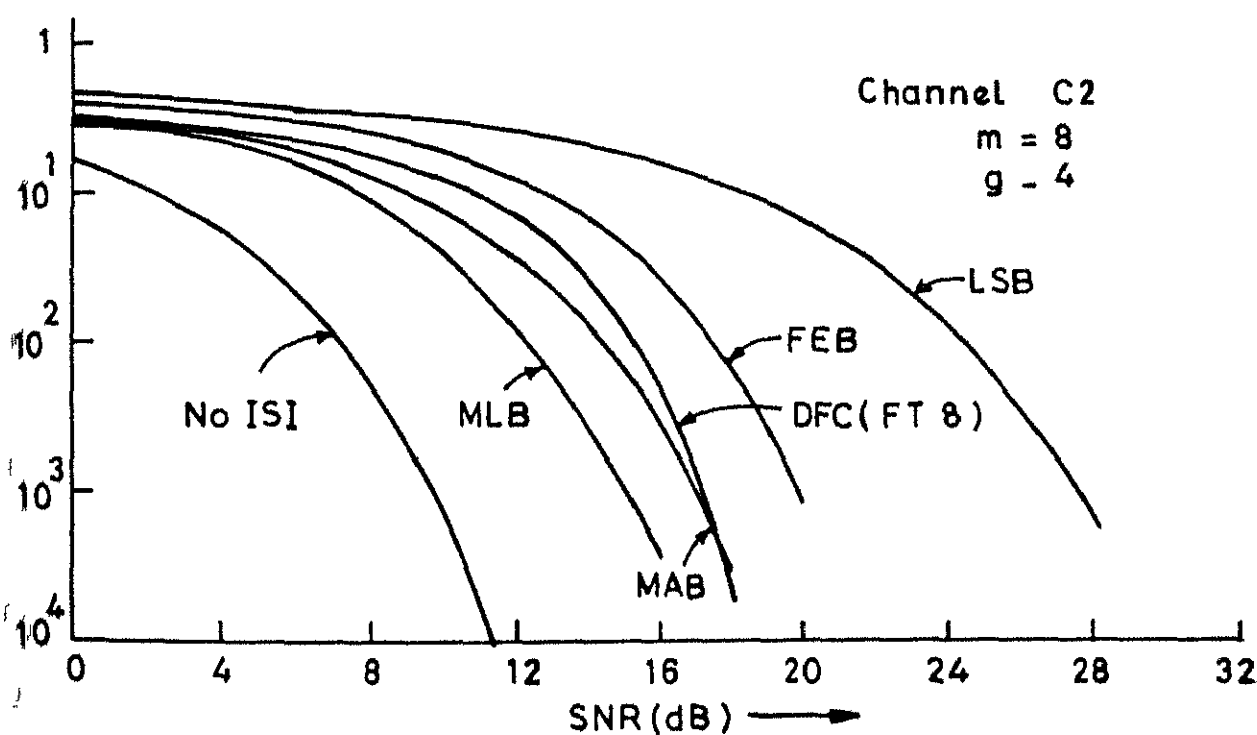
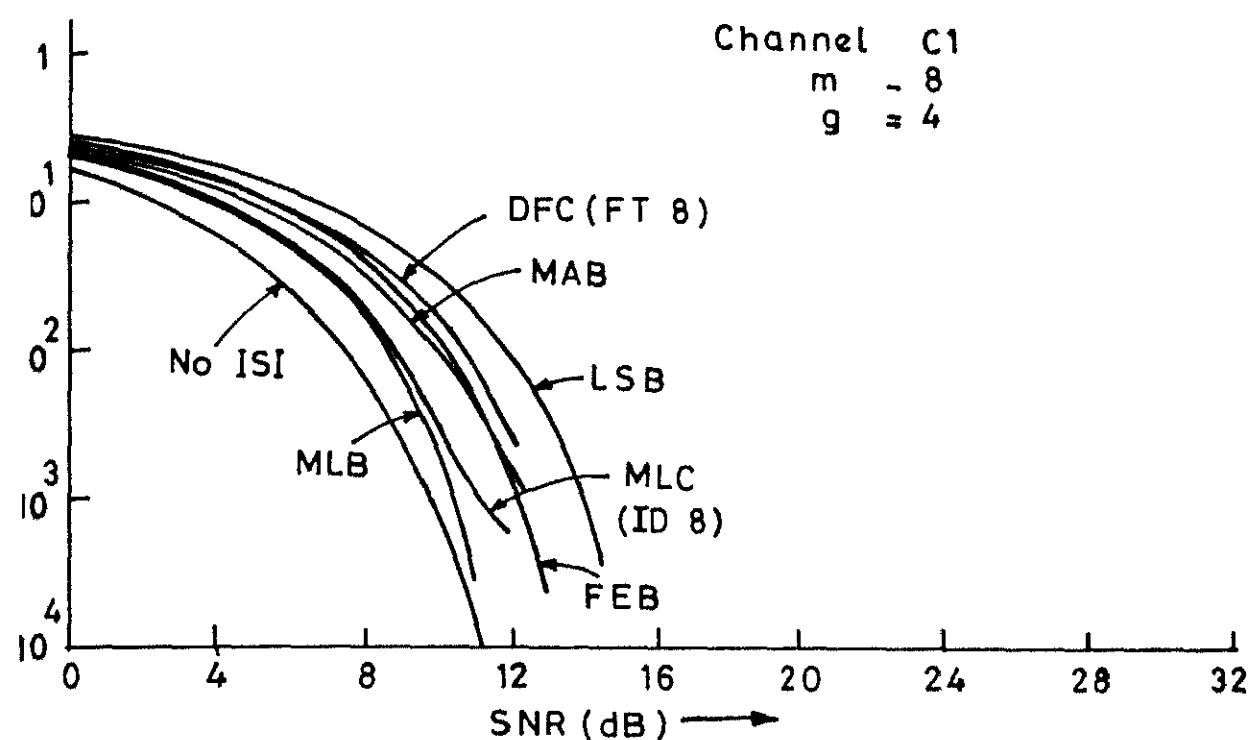


FIG 8.6 Comparison of performance of various DR scheme

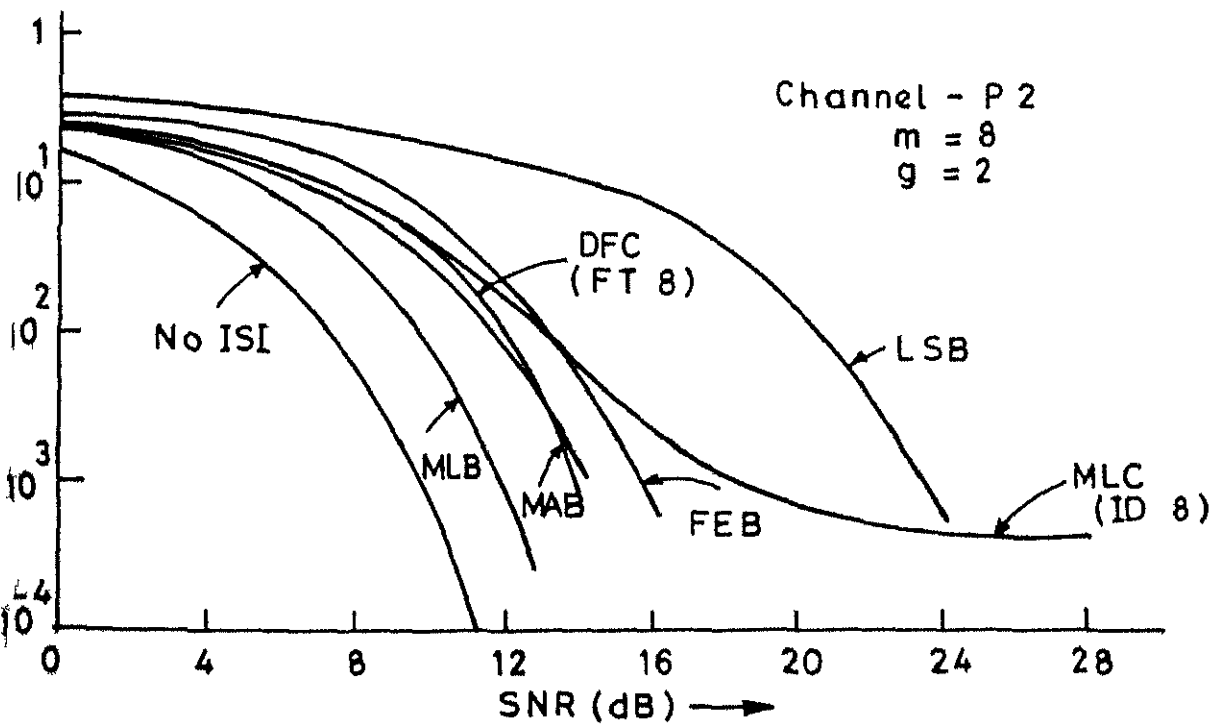
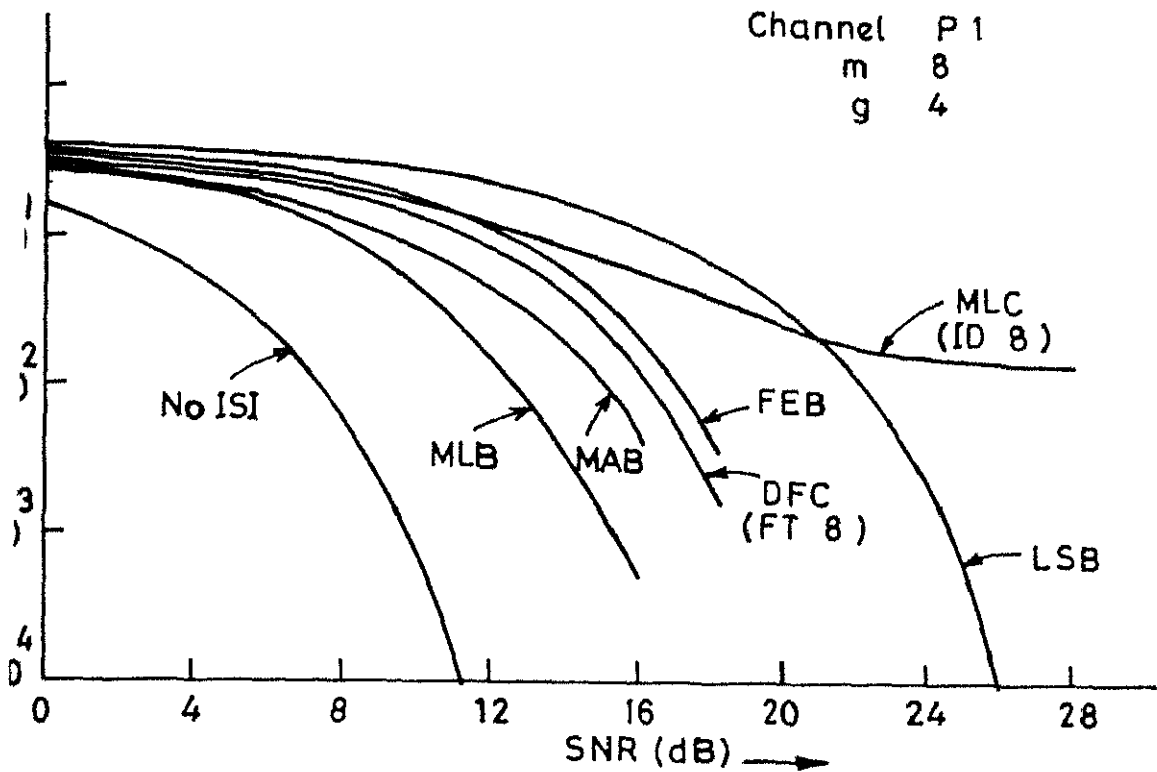


FIG 8 7 Comparison of performance of various DR schemes

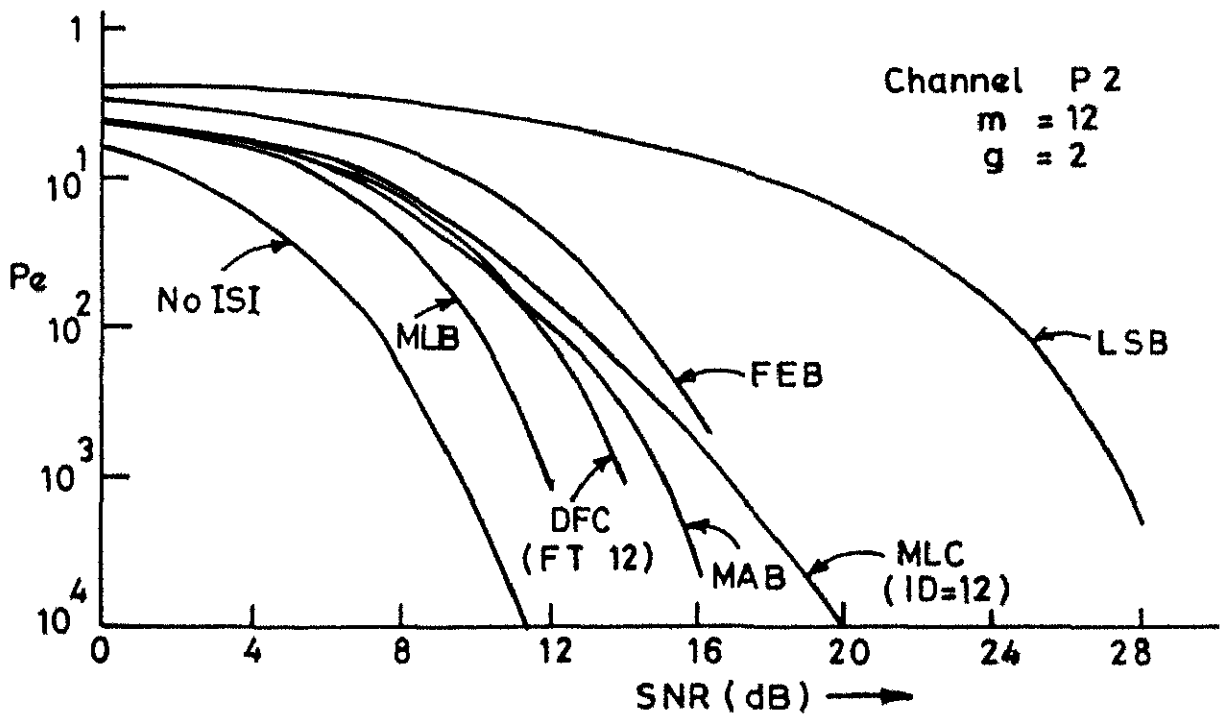
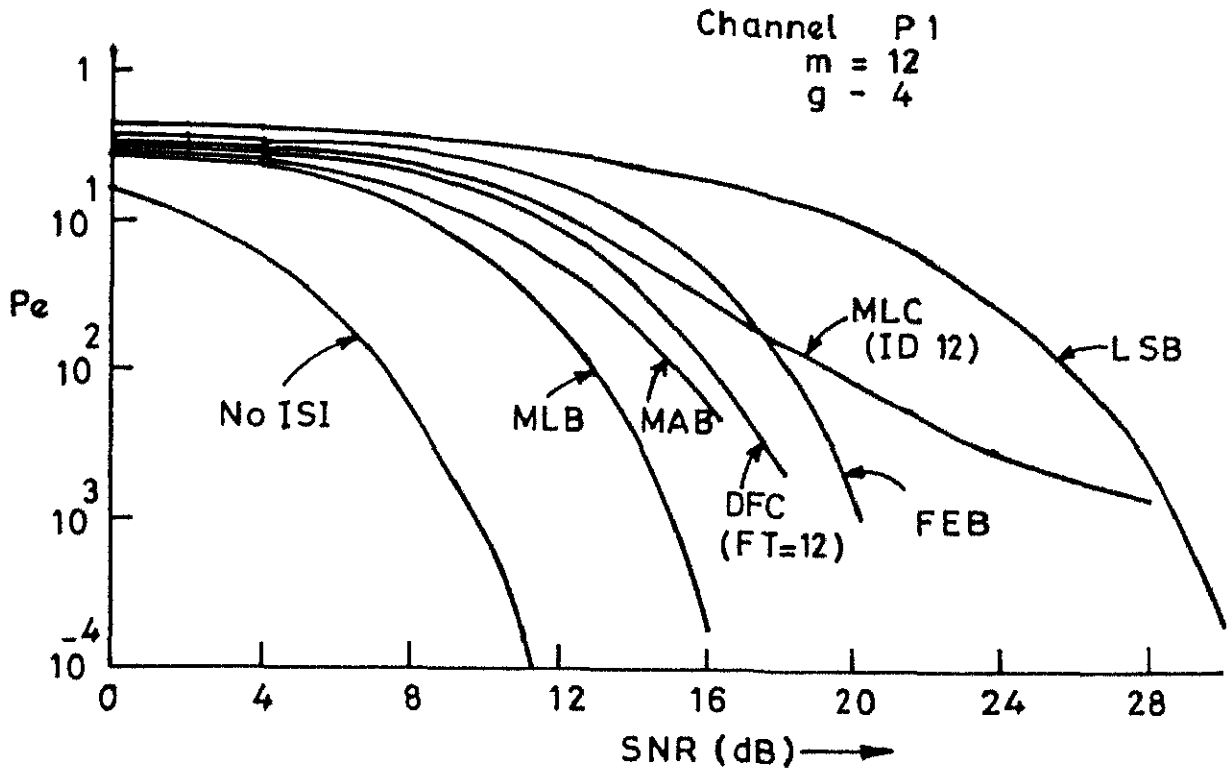


FIG 8 8 Comparison of performance of various DR schemes

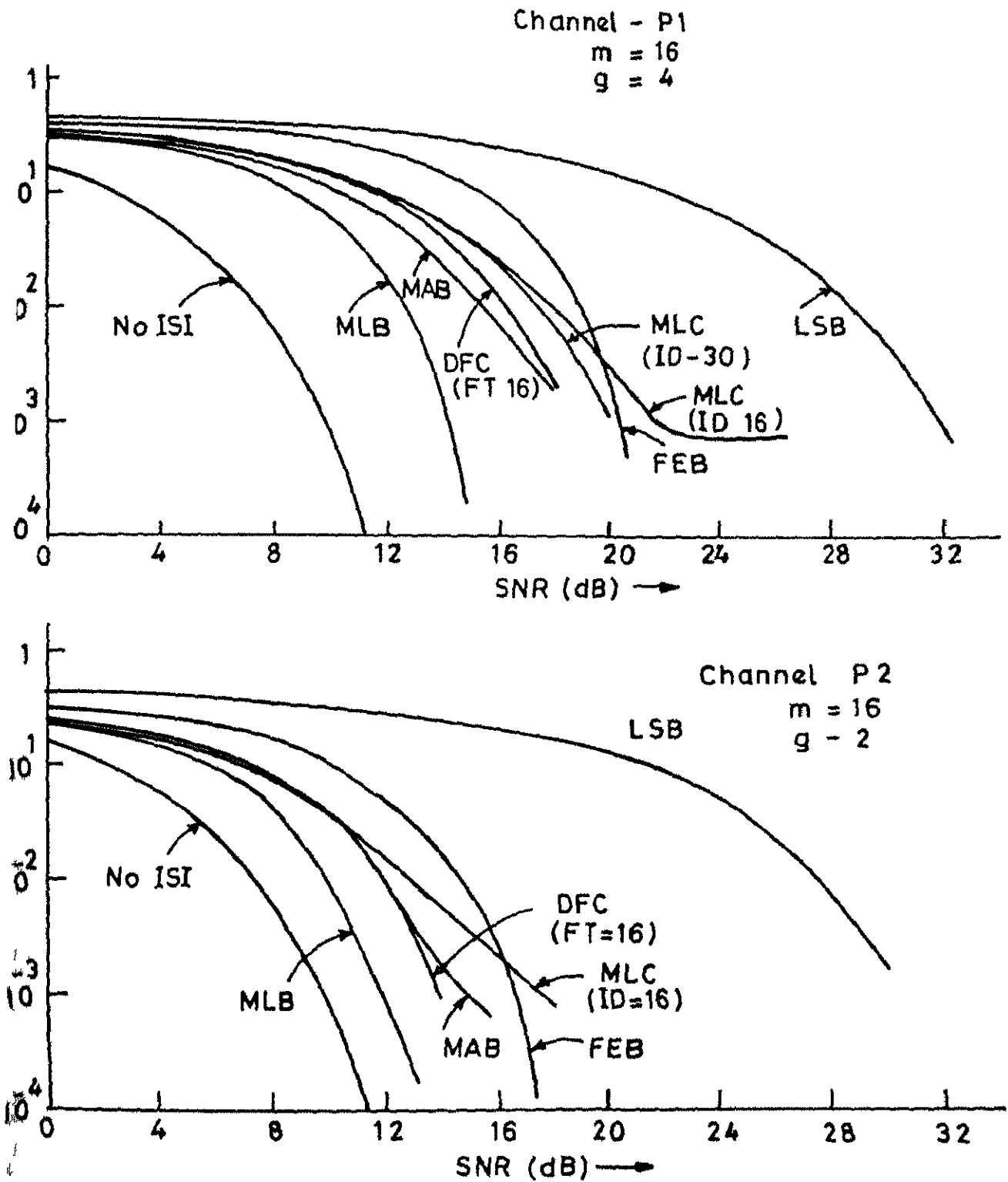
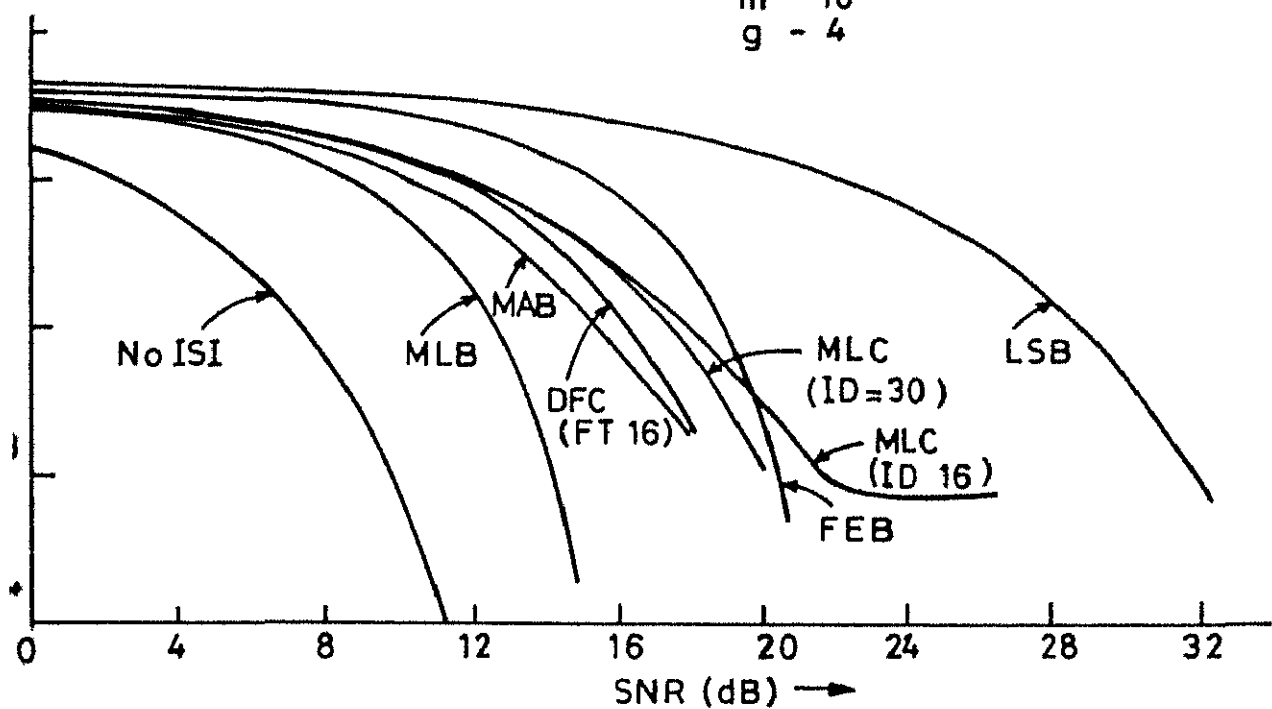
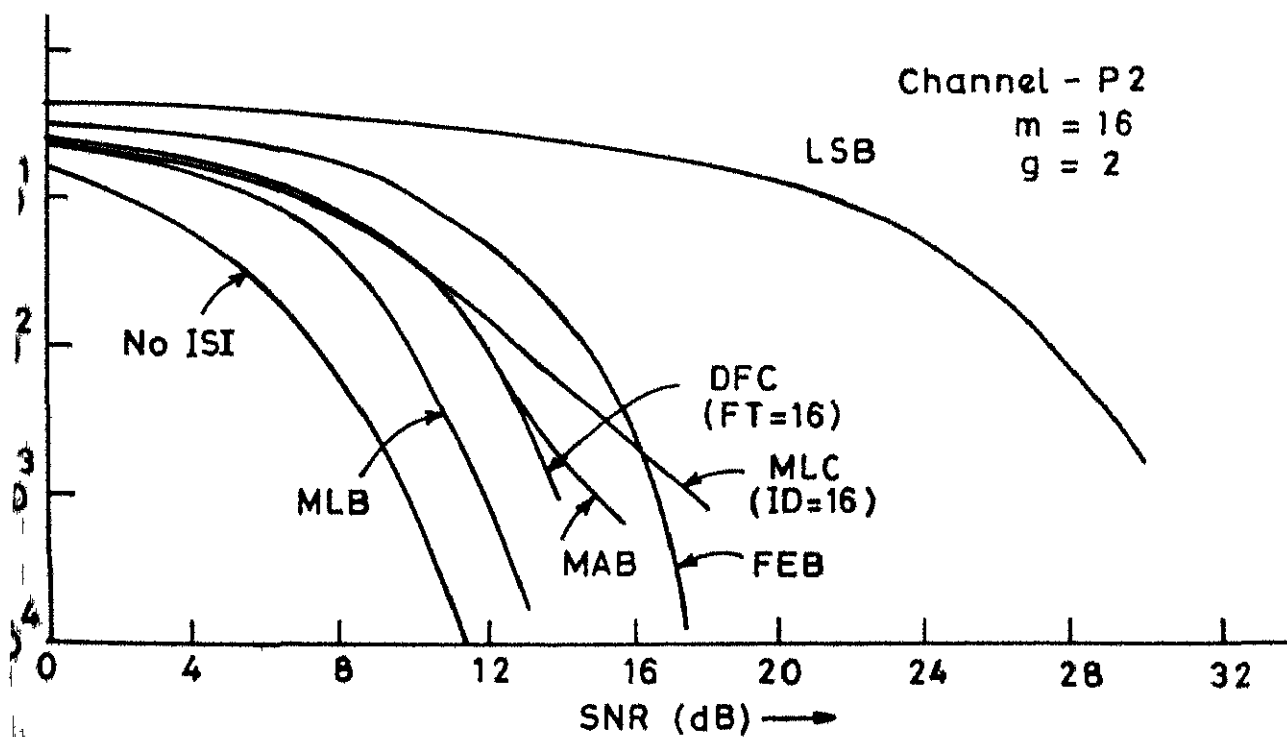


Fig. 8.9 Comparison of performance of various DR schemes

Channel P1
 $m = 16$
 $g = 4$



Channel - P2
 $m = 16$
 $g = 2$



5.8.9 Comparison of performance of various DR schemes

first stage. The procedure is continued till all the transmitted data are decided. The value of ID is mentioned in the figures. The decision feedback scheme for CDT (DFC) is based on the equations given in [7]. The number of forward taps used is denoted by T in the figures. For the sake of comparison error probability curve for channels with no ISI is also shown in the figures.

The Figures 8.1 to 8.9 cover a sufficiently wide range of block lengths and channel responses. The figures indicate that the performance of MLB is better than MLC when ID is equal to the block length. In the course of the simulation studies we have observed that for obtaining good performance at medium and large SNR values ID must be large in the case of MLC. In Figure 8.10 we have compared the performance of MLB and MLC on the maximum distortion channel $\frac{1}{\sqrt{5}}(1 \ 1 \ 1 \ 1 \ 1)$. Figure 8.10 shows that MLC performance saturates if ID is not large.

From the Figures 8.1 - 8.9 we also observe that MAB performs well compared to DFC when the block length is equal to the number of forward taps. However, it must be kept in mind that MAB requires more memory compared to DFC. It may be recalled that the computational complexity of both the schemes are more or less same. FEB, however, does not perform as well as MAB. One reason is that FEB assumes no knowledge of the noise variance. One attractive feature of FEB is that it can handle illconditioned channels more efficiently compared to the Trench algorithm.

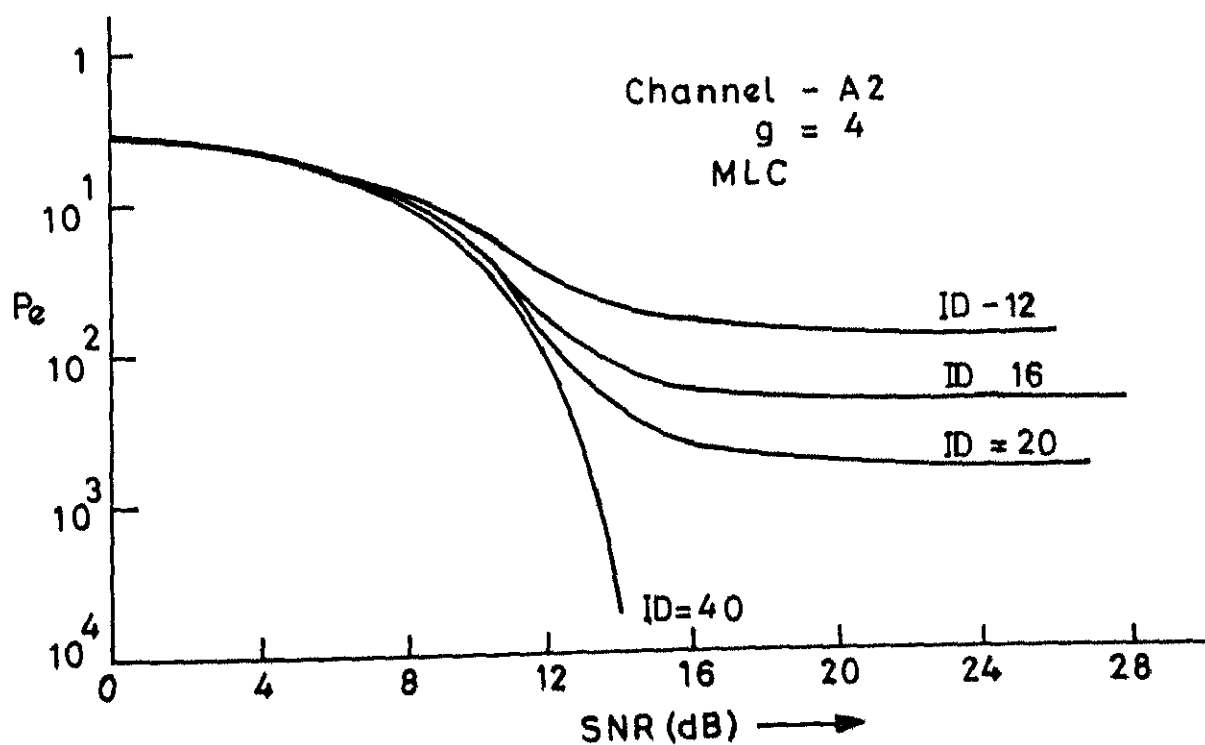
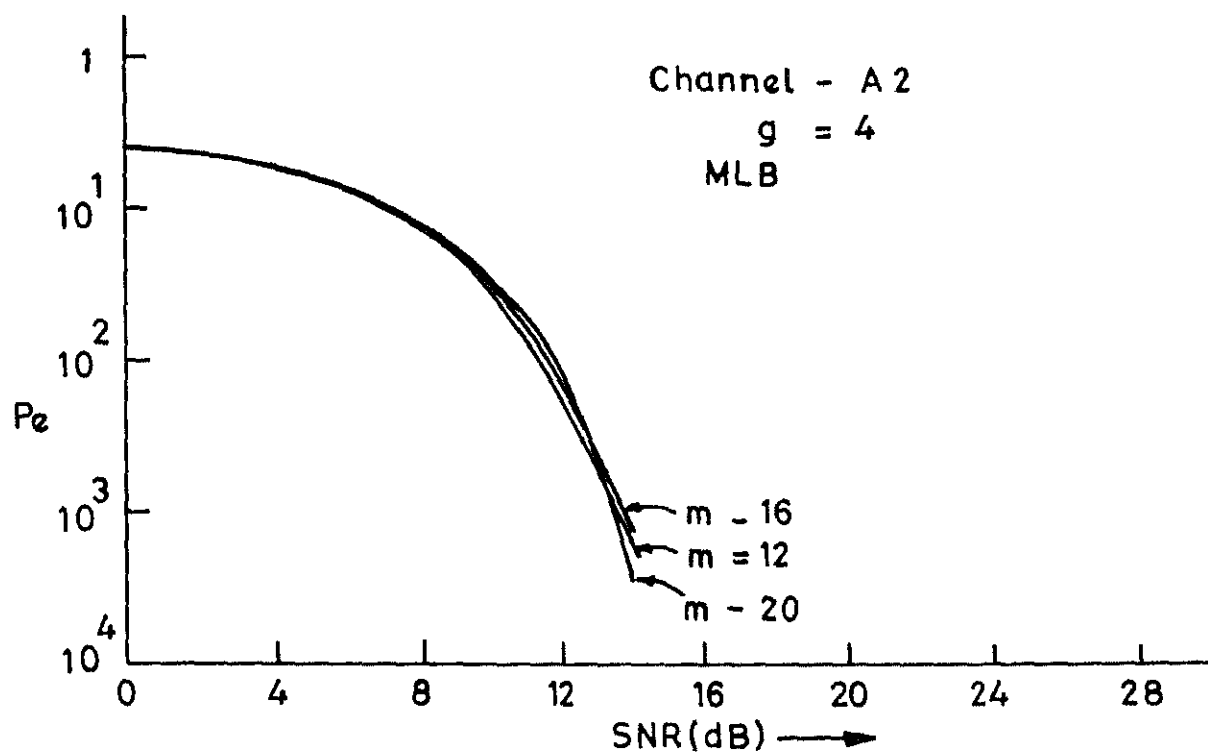


FIG 8 10 Comparison of MLB and MLC performance

based MAB in which illconditioned channels can give rise to large values of the tap coefficients

It is interesting to note that in the case of C1 (see Table 8.2) there is essentially no difference between MLC and MLB for block length of 8, 12, 16. In fact CCB performance is also very close to MLB. One reason for this perhaps is that d_{\min} for this channel is 1.93. Whereas for the other channels it is much less than 2 (see Table 8.2). In order to check the validity of this claim in Figure 8.11 we have plotted the performance curves for the example given in [33]. For this channel d_{\min} is 2. Performance curves indicate that the loss in performance due to CCB compared to MLB is not significant. This observation however needs more investigation.

Figure 8.12 shows the effect of block length on the performance of CCB and SCB in the case of channels C1 and P1. Although the performance in the case of channel C1 ($d_{\min}=1.93$) is better compared to that with P1 ($d_{\min}=1.13$) the performance depends strongly on block length.

8.3.1 The Effect of Increased ISI Among the Block Elements

So far we have considered BDT in which the effective data rate over the channel is less than the source data rate due to guard zeros following each block. We now consider the situation in which the data values of a block are transmitted at a higher

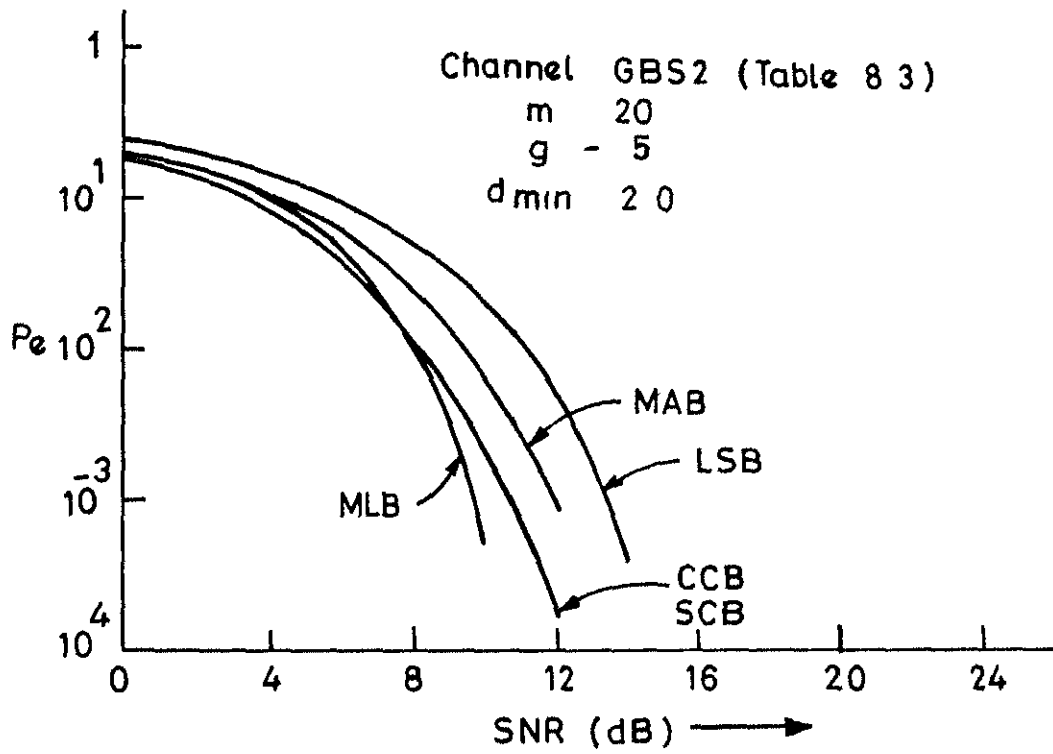


FIG 8.11 MMD Channel performance

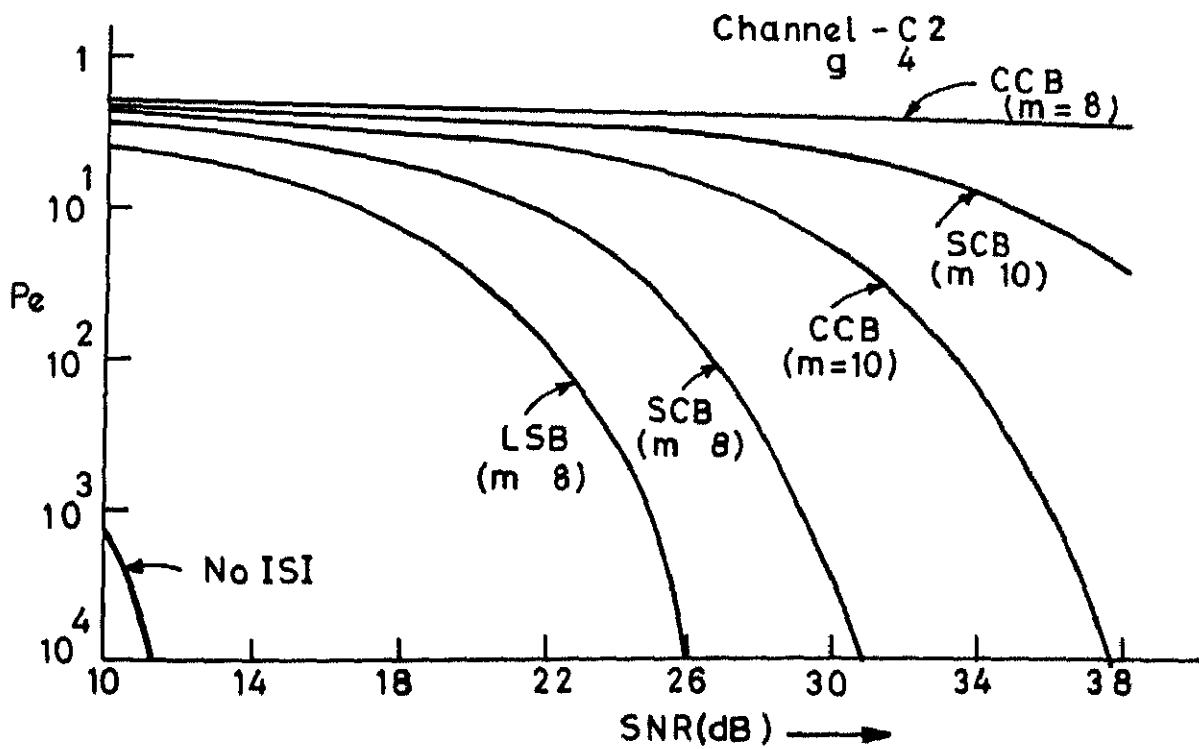
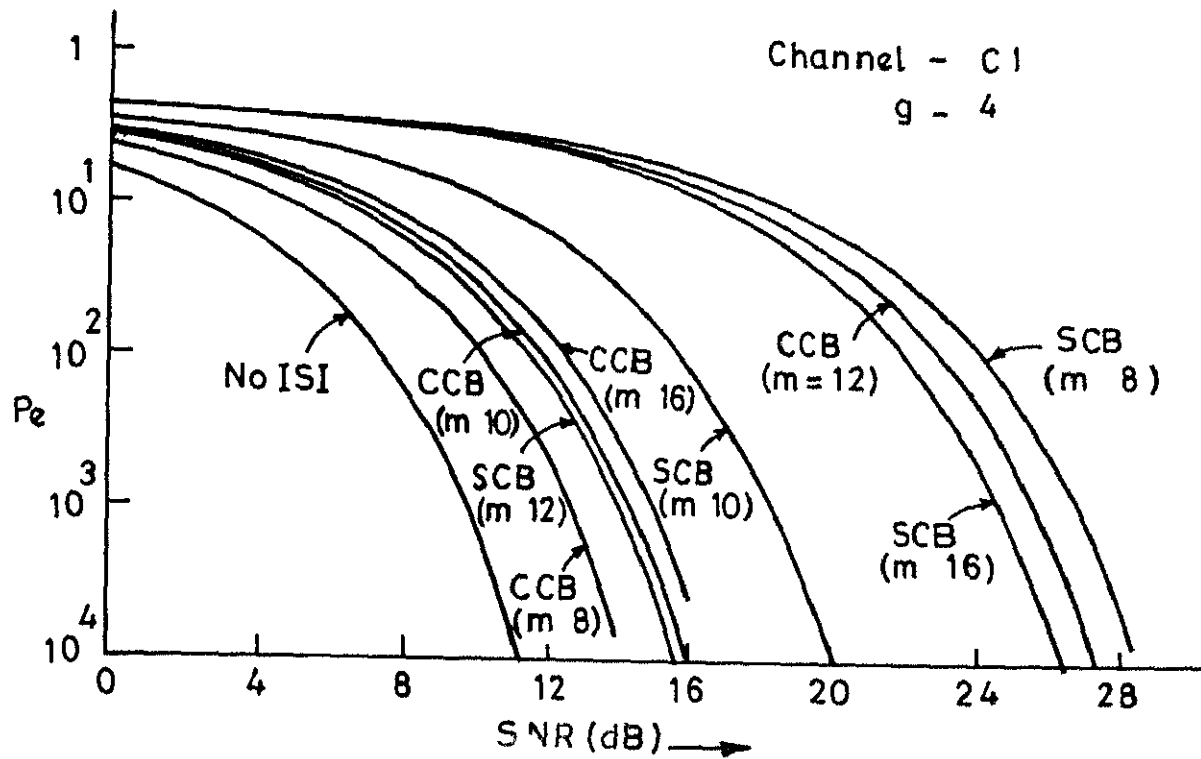


FIG 8 12 Effect of m on the performance of CCB and SCB

rate so that there is no reduction in the distortion over the channel. However, this results in increased ISI among the block members. The effect of increasing the data rate on the performance is the topic of this section.

First we consider maximum distortion channels. In Figure 8.13 we have shown the effect of increasing ISI when MLB is employed. For the sake of comparison we have also plotted MLC performance for ID equal to the block length. Note that the MLC curve for ID = 16 saturates after 14 dB.

Figure 8.14 shows the performance curves in the case of DFC and MAB. In this case also we observe that there is improvement in the performance of MAB compared to DFC.

As another example we considered a telephone channel given in [33]. In Table 8.3 we have given the equivalent

Table 8.3 Channel sampled response coefficients

Channel	Symbol duration (in nano-seconds)	Channel parameters	d_{min}
GBS1	3	0.13 - 0.67 0.74 4.03 9.10	2.0
GBS2	2.5	0.1 0.44 - 0.94 0.31 0.61 8.21	2.0
GBS3	2	0.17 0.44 - 0.86 2.72 6.76 6.78	1.719

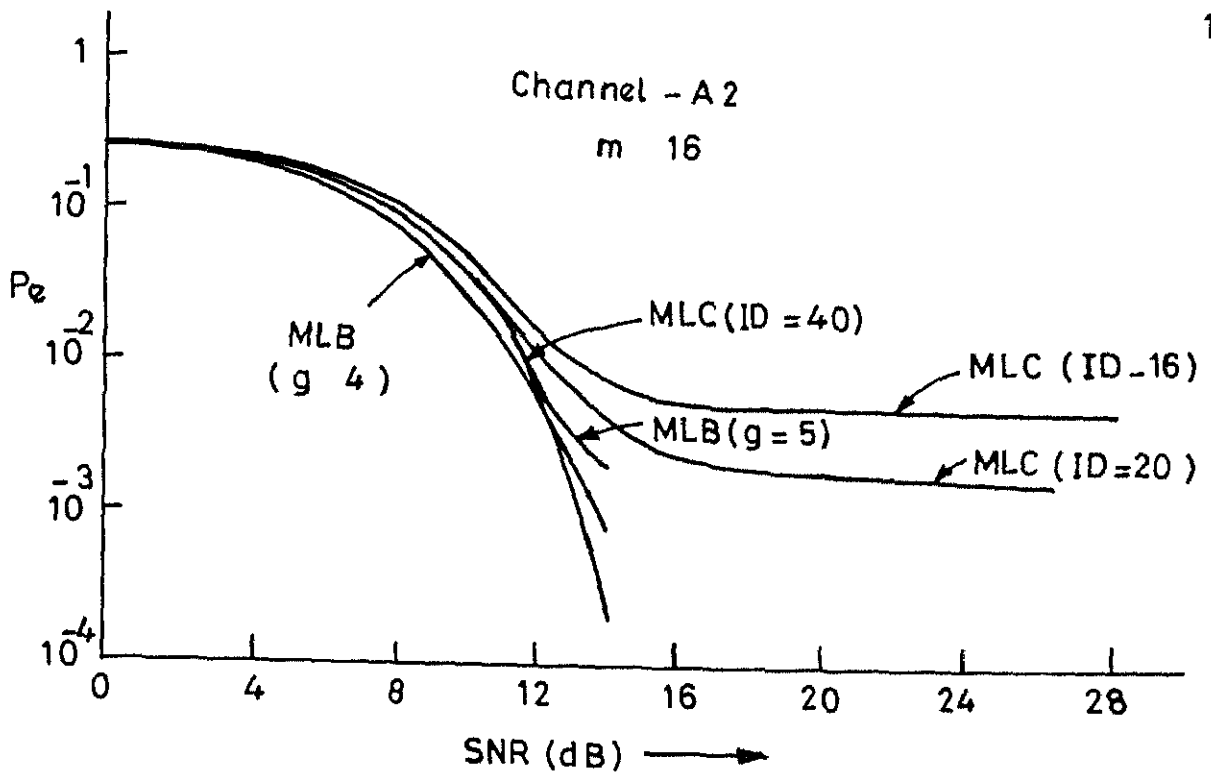


FIG 8 13 Effect of increased ISI on the performance of MLB

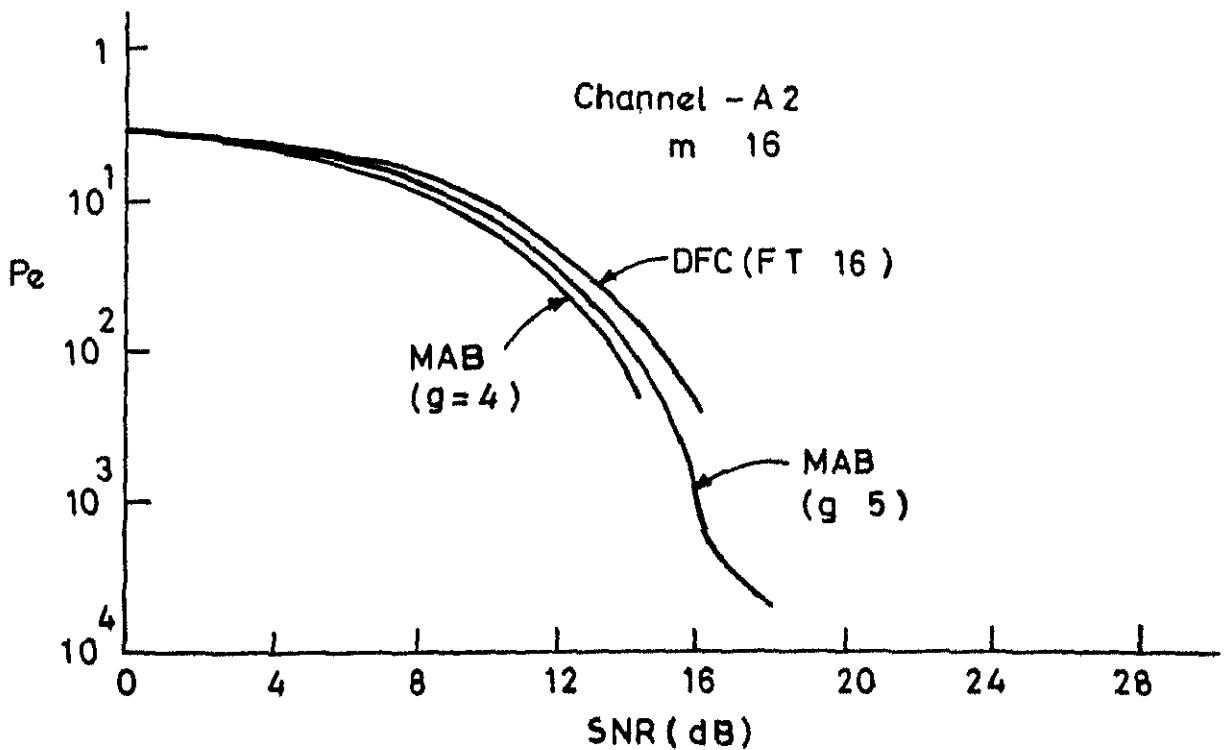


FIG 8 14 Effect of increased ISI on the performance of MAB

discrete time response values for various data rates. The performance curves are shown in Figures 8.1 to 8.16. In this case however BDT does not seem to enjoy noticeable advantage over CSDT.

Summarizing we evaluated the performance of various data recovery schemes using simulation. It may be noted that care must be exercised while making observations or while drawing conclusions regarding a particular scheme based on simulation results. Thus with limited number of examples and in the absence of any theoretical justification it is difficult to make any definite conclusion regarding the superiority of one scheme over the other. However from our limited simulation experiments on typical channels it appears that by choosing moderate block size and with increased data rate over the channel LB can be effectively implemented at a reasonable computational and storage complexity compared to MLC which requires large amount of storage as $ID \gg 5g$. Similarly we have observed that when the value of d_{min} is close to 2 CCB or SCB works well compared to MLB with relatively small amount of SNR loss. WAB also seems to be effective compared to DFC although it requires more storage. Moreover it limits error propagation to a block length in the worst situation - a feature absent in the case of DFC.

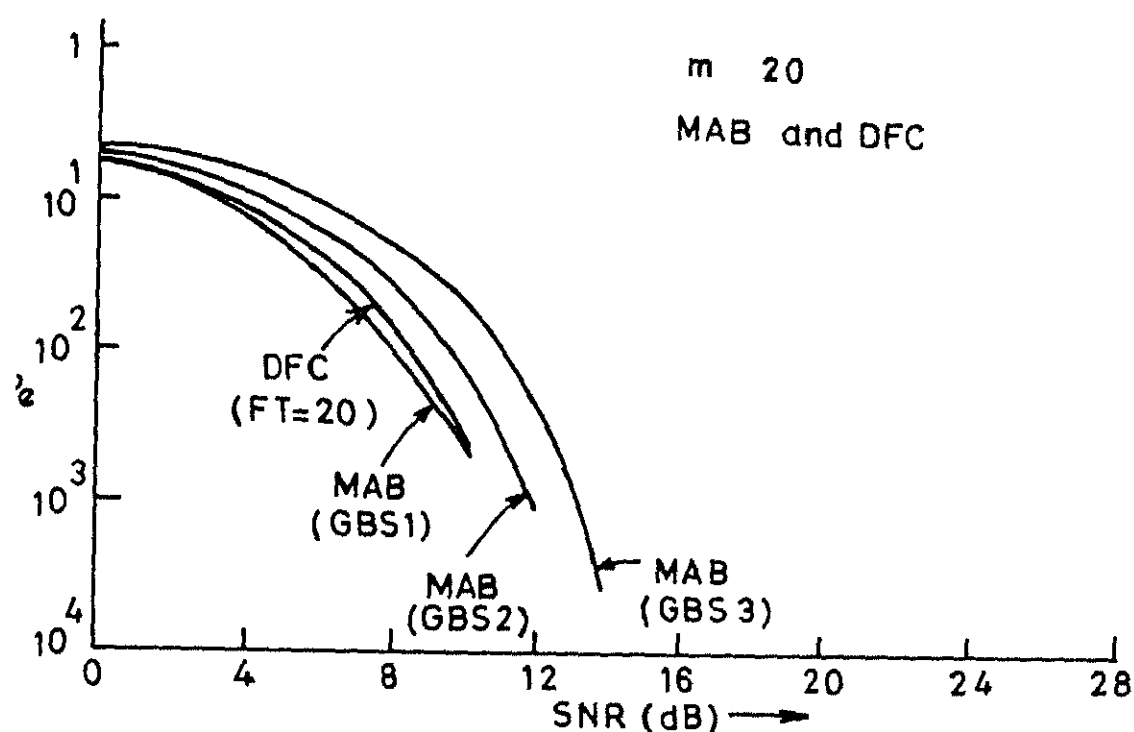


FIG. 8 15 Effect of increased ISI

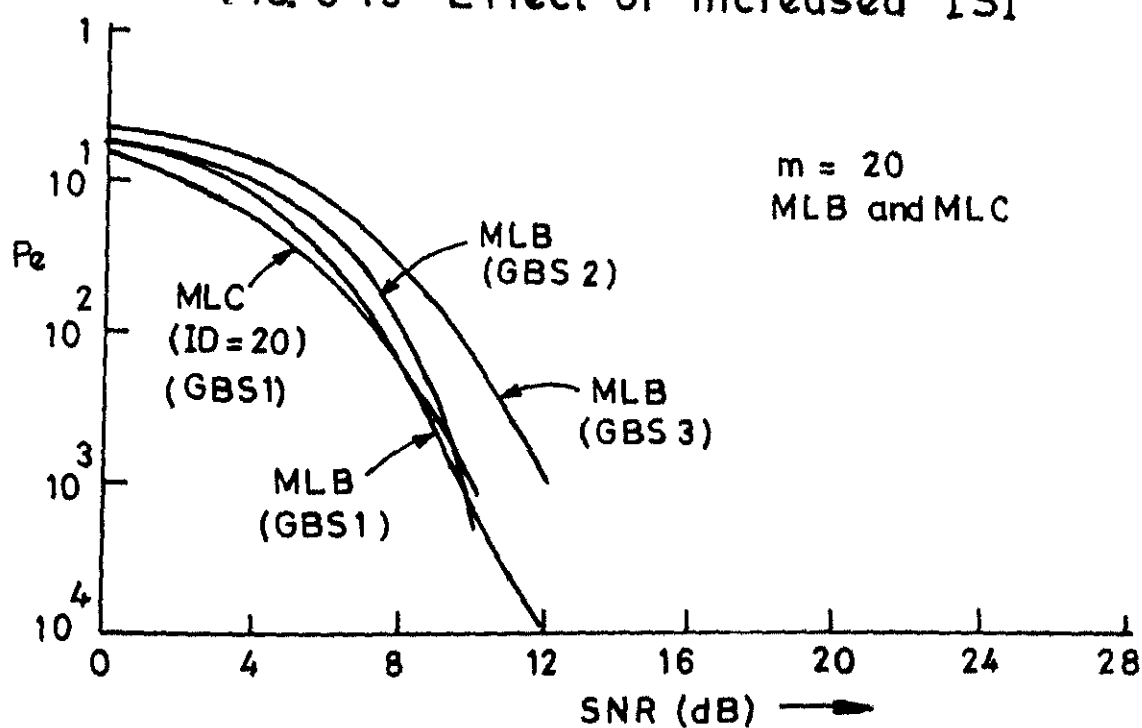


FIG. 8 16 Effect of increased ISI

CHAPTER 9

CONCLUSIONS AND SUGGESTIONS FOR FURTHER RESEARCH

In this thesis we have investigated the problem of recovering data in block data transmission systems. We have developed several data recovery schemes and compared their performance with DR schemes available for continuous serial data transmission systems. The data recovery problem has been viewed as a finite deconvolution problem in the presence of noise. Three deconvolution methods have been studied in detail. They are maximum likelihood decision feedback and least squares deconvolution methods. The DR schemes developed on the basis of these deconvolution methods may be viewed as counterparts of the maximum likelihood decision feedback and linear DR schemes in the context of CSDT.

At this stage it may be appropriate to note the following difference between BDT and CSDT systems. Due to practical considerations in CSDT systems truncated number of observations are used for making decision regarding a transmitted data symbol although in theory we must use the entire observation values. Thus the number of observations used for making decision determines the performance of a given DR scheme. In BDT we control the data rate and data transmission format so as to achieve better trade-off between error performance and computational complexity.

Salient features of the study carried out in this thesis and suggestions for further research are given in the following paragraphs

The maximum likelihood recursive convolutional algorithm developed in Chapter 3 is simpler to understand compared to the conventional Viterbi algorithm approach which makes use of Bellman's principle of optimality and state representation. The ML-RDA developed using Mitlen's lemma provides more insight into the data recovery problem compared to the Viterbi algorithm application. The computational complexity and storage requirements of ML-RDA grow linearly with the block length and exponentially with the channel memory. Simulation studies on some typical channels using ML-RDA for BDT and CSDT systems show that ML-RDA for CSDT system requires more storage and computational complexity compared to ML-RDA for BDT systems in order to achieve same error performance.

We have further shown that in the case of binary data, it is possible to reduce both storage and computational requirements of ML-RDA by using results from the theory of Boolean and pseudo-Boolean functions. Development of similar methods for multilevel data using some of the techniques given in [72] for dealing with functions of discrete variables is an interesting area for further research.

We have also shown that the method of minimizing pseudo Boolean functions provides a good framework for developing ML-decoding schemes for binary error correcting block codes. Examples given in Chapter 3 illustrate the usefulness of the method described. We however have not studied this aspect in detail. The technique developed in this thesis can be extended to nonbinary codes also. A detailed study of codes over other fields using some of the results available in [72] is expected to prove useful in a better understanding of ML decoding of linear as well as nonlinear block codes.

The decision feedback schemes developed in Chapter 4 are computationally more attractive in situations where ML-RDA becomes uneconomical. Unlike the conventional methods of deriving decision feedback data recovery schemes we have motivated the idea of decision feedback via Gaussian elimination method. We have shown that the modified Austin approach can also be viewed as Gaussian elimination method with a nonlinear decision device.

The DF-DR schemes for BDT systems restrict error propagation to at most a block length whereas in the case of DF-DP schemes for CSDT systems error propagation can extend to much longer durations. However DR schemes for BDT systems require more storage compared to DR schemes for CSDT systems for achieving same probability of error performance.

The DR scheme studied in Chapter 5 have the least computational and storage requirements compared to those studied in Chapters 3 and 4. However this is only at the expense of probability of error performance. However simulation studies show that in the case of a good number of channels DR schemes based on k -Circulant completions perform well compared to DR schemes based on LS-deconvolution. We have shown that k -Circulant completions of the channel convolution matrix provides a more general setting for studying deconvolution problems using discrete Fourier transform. A detailed study of optimum choice of k -Circulant completions according to some performance criteria needs further investigation.

In Chapter 6 we have showed that deconvolution using DFT and least squares deconvolution can be viewed as particular cases of square completions of the channel convolution matrix.

Detailed analysis of the Steepest descent algorithm in Chapter 7 shows that in the case of some examples good estimates of the convergence parameter can be obtained using k -Circulant completions. This analysis clearly explains the observation made by others that DFT domain fixed step steepest descent algorithm converges faster than the sample domain schemes. Moreover we have shown that such estimates can be used in the sample domain to achieve good convergence.

The simulation results given in Chapter 8 cover a wide range of block lengths and typical channel examples. It is interesting to note that the effect of increased ISI among the symbols of a block due to increased data rate over the channel on the performance is not significant. Simulation studies also indicate that with appropriate choice of block lengths BDT provides better trade-off between error performance and processing complexity compared to the available BDR schemes for CSDT systems. However, as simulation results are always based on a few examples, no definite claims can be made regarding the superiority of BDT over CSDT.

The maximum minimum distance and maximum distance channels obtained in Chapter 8 show that some of the good channels are partial response channels. However, in the absence of any clear definition of partial response channel [94] it is difficult to show the connection between partial response and MDD or MD channels. This aspect needs further investigation.

Another aspect of the data recovery problem in which we could not achieve much success is in obtaining good bounds for the error performance of maximum likelihood and decision feedback schemes. The difficulty seems to be due to the nonlinear nature of the BDR schemes. However, we still feel that it must be possible to obtain useful bounds for probability of error which in this case will be functions of block length and channel

memory this aspect requires further investigation. Some work in this direction has been carried out by Klein and Wolf [54].

In this thesis we have not considered the problem of adapting the parameters of the data recovery schemes to the channel variations. Depending on the nature of the channel variation the design of the data recovery schemes will change. The problem is interesting due to the presence of noise and our limitation in recovering data without errors. Further investigation is needed to understand various aspects of this problem.

APPENDIX A

MITTEN'S LEMMA

In this appendix we state and prove Mitten's lemma [55]

Mitten's Lemma Let $X = \{x\}$ be an arbitrary set and let $Y_1(x) = \{y_1\}$ $i = 1, 2, \dots, n$ be a finite sequence of arbitrary sets whose definitions may depend on $x \in X$. Let $g_1(x, y_1)$ $i = 1, 2, \dots, n$ be a sequence of real valued functions for which $g_1^*(x) = \min[g_1(x, y_1) | y_1 \in Y_1(x)]$ exists for $i = 1, 2, \dots, n$ and all $x \in X$. Let $f(x)$ be an arbitrary function and let $g(x, y_1, \dots, y_n) = h[f(x), g_1(x, y_1), \dots, g_n(x, y_n)]$ be a real valued function for which $g^* = \min[g(x, y_1, \dots, y_n) | x \in X, y_1 \in Y_1(x), \dots, y_n \in Y_n(x)]$ exists and $h^*(x) = h[f(x), g_1^*(x), \dots, g_n^*(x)]$ exist for all $x \in X$. If $h(x) \leq g(x, y_1, \dots, y_n)$ for every $x \in X$ and $y_1 \in Y_1(x), \dots, y_n \in Y_n(x)$ then $g^* = \min[h^*(x) | x \in X] = h^*$.

Proof The lemma can be proved by the method of contradiction. For this purpose assume that there exists $x \in X$ and $y_1 \in Y_1(x), \dots, y_n \in Y_n(x)$ such that $g(x, y_1, \dots, y_n) < h^*$.

But from the definition of minimum $h^* \leq h^*(x)$ and $h(x) \leq g(x, y_1, \dots, y_n)$ by the assumed property of $h^*(x)$. Therefore

$$g(x, y_1, \dots, y_n) < h^* \leq h^*(x) < g(x, y_1, \dots, y_n)$$

which is a contradiction.

APPENDIX B

k-CIRCULANT MATRICES

The purpose of this appendix is to list a few properties of k-Circulant matrices [81] [82]

The k-Circulant matrices considered in our study have the following structure

$$C_k = \begin{bmatrix} C_0 & kC_{n-1} & kC_{n-2} & \dots & kC_1 \\ C_1 & C_0 & kC_{n-1} & \dots & kC_2 \\ C_2 & C_1 & C_0 & \dots & kC_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ C_{n-1} & C_{n-2} & C_{n-3} & \dots & C_0 \end{bmatrix}$$

where $|k| = 1$. When $k = 1$, C_1 is a Circulant matrix. $k = -1$ gives skew Circulant matrix. Skew Circulants are also known as negacyclic matrices [82]

It is convenient to study k-Circulant with the help of the simple k-Circulant matrix

$$\eta_k = \begin{bmatrix} 0 & kI_1 \\ I_{n-1} & O_{n-1, n-1} \end{bmatrix}$$

A matrix A is said to be r Circulant iff

$$A\eta_k = \eta_k A$$

In terms of η_k C_k can be expressed as

$$C_k = C_0 I_n + C_1 \eta_k + C_2 \eta_k^2 + \dots + C_{n-1} \eta_k^{r-1} \quad (B.1)$$

Since every k Circulant matrix commutes with η_k it is sufficient to know the eigenvalues and eigenvectors of η_1 in order to characterise and analyse Circulants.

The eigenvalues of η_k are the n th roots of k for the characteristic equation of η_k is given by

$$\lambda^n - k = 0$$

Let p be an n th root of k . Then $p\omega^l - 1 = 0$ $l = 0, 1, \dots, n-1$

where $\omega = e^{j2\pi/n}$ represents n roots of $\lambda^n - k = 0$

We now show that the vector

$$x = (1, x^{-1}, x^{-2}, \dots, x^{-(n-1)})^T \text{ where } x \text{ is an } n\text{th}$$

root of k is an eigenvector of η_k

A simple calculation shows that

$$\eta_k x = \begin{bmatrix} 1 & x^{-(n-1)} \\ x^{-1} & \\ x^{-2} & \\ \vdots & \\ x^{-(n-2)} & \end{bmatrix} = x \underline{x}$$

Thus n roots of k generate n eigenvectors of η_k . Let F_k^{-1} be a matrix with the n eigenvectors of η_k as columns. It has the following form

$$F_k^{-1} = \frac{1}{\sqrt{n}} \begin{bmatrix} 1 & 1 & 1 \\ p^{-1} & (p\omega)^{-1} & (p\omega^{n-1})^{-1} \\ p^{-2} & (p\omega)^{-2} & (p\omega^{n-1})^{-2} \\ \vdots & \vdots & \vdots \\ p^{-(n-1)} & (p\omega)^{-(n-1)} & (p\omega^{n-1})^{-(n-1)} \end{bmatrix} \quad (B.2)$$

Note that $\frac{1}{\sqrt{n}}$ is just a normalization factor used for making $F_k^{-1} \Gamma_k = I_n$

Using B.2 it can be shown that

$$F_k \eta_k F_k^{-1} = \Lambda_k \quad (B.3)$$

where Λ_k is a diagonal matrix with $p\omega^{i(n-1)} = 0, 1, \dots, n-1$ as diagonal elements

When $k=1$, $F_k = F_1 = F$ the DFT matrix is obtained. In virtue of this we shall refer to F_k as k -DFT matrix. In terms of F we can write F_k^{-1} as

$$F_k^{-1} = D^{-1} F^{-1} \quad (B.4)$$

where D is a diagonal matrix with $p^{i(n-1)} = 0, 1, \dots, n-1$ as diagonal entries. Using (B.4) in (B.3) we obtain

$$F D \eta_k D^{-1} F^{-1} = \Lambda_k \quad (B 5)$$

If C_1 is a Circulant matrix we know that

$$F C_1 F^{-1} = \Lambda_1 \quad (B 6)$$

where Λ_1 is a diagonal matrix with eigenvalues of C_1 as diagonal entries

Comparing (B 6) and (B 5) we see that $D \eta_k D^{-1}$ represents a Circulant matrix

Post and pre multiplying B 1 by F_k^{-1} and F_k respectively and using (B 3) we obtain

$$F_k C_k F_k^{-1} = C_0 I + C_1 \Lambda_k + C_2 \Lambda_k^2 + \dots + C_{n-1} \Lambda_k^{n-1} \quad (B 7)$$

From (B 7) we see that if $\lambda_1 = 0, 1, \dots, n-1$ are the eigenvalues of C_k

$$\lambda_1 = \sum_{j=0}^{n-1} C_j (p\omega^1)^j \quad (B 8)$$

(B 8) can also be rewritten as

$$\lambda_1 = \sum_{j=0}^{n-1} (C_j p^j) \omega^{1j} \quad (B 9)$$

(B 9) exhibits an interesting aspect eigenvalues of C_k can be determined by finding the eigenvalues of a Circulant matrix having first column

$$(C_0 \ C_1 p \ C_2 p^2 \ \dots \ C_{n-1} p^{n-1})^T \quad (B 10)$$

This can also be seen directly using (B 7). Using (B 4) in (B 7) we get

$$FDC_k D^{-1} \Gamma^{-1} = \Lambda = \text{diag}(\lambda_0, \lambda_1, \dots, \lambda_{n-1}) \quad (\text{B } 11)$$

Therefore $DC_k D^{-1}$ is a Circulant matrix. By little calculation we can see that the first column of this matrix is given by (B 10)

Finally consider inverse of a k -Circulant matrix (B 3) in terms of η_k^{-1} whenever η_k^{-1} exists. We get

$$\eta_k^{-1} = \Gamma_k^{-1} \Lambda_k^{-1} \Gamma_k$$

or
$$F_k \eta_k^{-1} F_k^{-1} = \Lambda_k^{-1}$$

which shows that inverse of a k -Circulant matrix whenever it exists is also a k -Circulant.

R.L.F. R. NCES

- 1 Nyquist H Cert in topics in telegraph transmission theory Trans AIEE vol 47 1928 pp 617-644
- 2 Clark A P Adaptive detection of distorted digital signals Radio Electron Eng vol 40 1970 pp 107-119
- 3 Clark A P A synchronous serial data transmission system using orthogonal groups of binary signal elements IEE Trans Commun vol COM-19 1971 pp 1101-1110
- 4 Clark A P Advanced data transmissions ns Pentech Press London 1977
- 5 Bennet / R and Davey J R Data transmission McGraw-Hill New York 1965
- 6 Lucky R W Salz J and Weldon L J Jr Principles of data communications McGraw-Hill New York 1968
- 7 Proakis J G Digital communications McGraw Hill Tokyo 1983
- 8 Franks L E Data communication Fundamentals of baseband transmission Dowden Hutchinson and Ross Inc Pennsylvania 1974
- 9 Lucky R W A survey of the communication theory literature 1968-1973 IEE Trans Inform Theory vol IT-19 1973 pp 725-739
- 10 Tufts D W Nyquist's problem -The joint optimization of transmitter and receiver in pulse amplitude modulation Proc IEEE vol 53 1965 pp 248-259
- 11 Tufts D W and Shnidman D A Optimum waveforms subject to both energy and peak value constraints Proc IEEE vol 52 1964 pp 1002-1007
- 12 Smith J W The joint optimization of transmitted signal and receiving filter for data transmission systems Bell Syst Tech Journ vol 44 1965 pp 2303-2392

- 13 Austin M E Decision feedback equalization for digital communication over dispersive channels MIT Lincoln Lab Lexington Mass Tech Rept 437 1967
- 14 Lucky R W Automatic equalization for digital communication Bell Syst Tech Journ vol 44 1965 pp 47-588
- 15 Sha R T and Tang D T A new class of automatic equalizers IBM J Res Dev vol 16 1972 pp 556-566
- 16 Chang R W and Ho E Y On fast start up data communication systems using pseudo-random training sequences Bell Syst Tech Journ vol 51 1972 pp 2013-2022
- 17 Mueller K H and Spaulding D A Cyclic equalization - a new rapidly converging equalization technique for synchronous data communication Bell Syst Tech Journ vol 54 1975 pp 369-406
- 18 Kosovych O S and Picholtz Automatic equalization using successive overrelaxation iterative technique IEEE Trans Inform Theory vol IT 21 1975 pp 51-58
- 19 Chang R W A new equalizer structure for fast start up digital communication Bell Syst Tech Journ vol 50 1971 pp 1969-2014
- 20 Rudin H R Automatic equalization using transversal filters IEEE Spectrum vol 4 1967 pp 53-59
- 21 Walzman T and Schwartz M Automatic equalization using the discrete frequency domain IEEE Trans Inform Theory vol IT-19 1973 pp 59-68
- 22 Walzman T and Schwartz M A projected gradient method for automatic equalization in the frequency domain IEEE Trans Commun vol COM-21 1973 pp 1442-1446
- 23 Lucky R W Techniques for adaptive equalization of digital communication systems Bell Syst Tech Journ vol 45 1966 pp 255-286
- 24 Gersho A Adaptive equalization of highly dispersive channels for data transmission Bell Syst Tech Journ vol 48 1969 pp 55-70
- 25 Proakis J G and Miller J H An adaptive receiver for digital signalling through channels with intersymbol interference IEEE Trans Inform Theory vol IT 15 1969 pp 484-497

- 26 Di Toro M J Communication in time frequency spread media using adaptive equalization I E Proc vol 56 1968 pp 1653-1679
- 27 Hirsch D and Wolf W J A simple adaptive equalizer for efficient data transmission IEEE Trans Commun vol COM-18 1970 pp 5-12
- 28 Schonfeld T J and Schwartz M Rapidly converging second order tracking algorithm for adaptive equalization IEEE Trans Inform Theory vol IT-17 1971 pp 572-579
- 29 Salz J Optimum mean-square decision feedback equalization Bell Syst Tech Journ vol 52 1973 pp 1341-1373
- 30 Taylor D P Nonlinear feedback equalizer employing a soft limiter Electronics Letters vol 7 1971 pp 265-267
- 31 Taylor D P The estimate feedback equalizer a sub optimum nonlinear receiver IEEE Trans Commun vol COM-21 1973 pp 979-990
- 32 Monsen P Feedback equalization for fading dispersive channels IEEE Trans Inform Theory vol IT-17 1971 pp 56-65
- 33 George D A Bowen R P and Storcy J P An adaptive decision feedback equalizer IEEE Trans Commun vol COM-19 1971 pp 281-293
- 34 Duttweiler D L Mazo J E and Messerschmitt D C An upper bound on the error probability in decision feedback equalization IEEE Trans Inform Theory vol IT-20 1974 pp 490-497
- 35 Salazar A C Design of transmitter and receiver filters for decision feedback equalization Bell Syst Tech Journ vol 53 1974 pp 503-523
- 36 Lkanayake N and Taylor P A decision feedback receiver structure for bandlimited nonlinear channels IEEE Trans Commun vol COM-29 1981 pp 539-548
- 37 Ling F and Proakis J G Adaptive lattice decision feedback equalizers their performance and application to time invariant multipath channels IEEE Trans Commun vol COM-33 1985 pp 348-356

- 38 Forney G D Jr Maximum likelihood sequence estimation of digital sequences in the presence of intersymbol interference IEEE Trans Inform Theory vol IT-13 1972 pp 363-378
- 39 Viterbi A J Error bounds for convolutional codes and an asymptotically optimum decoding algorithm IEEE Trans Inform Theory vol IT-13 1967 pp 201-269
- 40 Viterbi A J and Omura J K Error bounds for intersymbol interference channels CISM Lectures No 229 The information theory approach to communications Edited by G Longo Springer Verlag 1978
- 41 Hayes J F The Viterbi algorithm applied to digital transmission IEEE Communications Society vol 13 1975 pp 15-20
- 42 Forney G D Jr The Viterbi algorithm Proc IEEE vol 61 1973 pp 268-278
- 43 Kobayashi H A survey of coding schemes for transmission or recording of digital data IEEE Trans Commun vol COM-19 1971 pp 1087-1100
- 44 Qureshi S U H and Newhall E E An adaptive receiver for data transmission over time dispersive channels IEEE Trans Inform Theory vol IT-19 1973 pp 448-457
- 45 Fredricsson S A Joint optimization of transmitter and receiver filters in digital PAM systems with a Viterbi detector IEEE Trans Inform Theory vol IT-22 1976 pp 200-209
- 46 Lee W U and Hill F S Jr A maximum likelihood sequence estimator with decision feedback equalization IEEE Trans Commun vol COM-25 1977 pp 971-980
- 47 Falconer D D and Magee F R Adaptive channel memory truncation for maximum likelihood sequence estimation Bell Syst Tech Journ vol 52 1973 pp 941-945
- 48 Fredricsson S A Optimum transmitting filter in digital PAM systems with a Viterbi detector IEEE Trans Inform Theory, vol IT-20 1974 pp 479-489
- 49 Antonio C and Kong K Further results on Viterbi equalizer IEEE Trans Inform Theory vol IT-20 1974 pp 764-767

- 50 Beare C T The choice of the desired impulse response in combined linear Viterbi algorithm equalizers IEEE Trans Commun vol COM-26 1978 pp 1301-1307
- 51 Kazakos D Computational savings and implementation of maximum likelihood detectors IEEE Trans Inform Theory vol IT-24 1978 pp 124-126
- 52 Clark A P and Clayden M Pseudo binary Viterbi detector IEEE Proc vol 131 No 2 April 1984 pp 208-218
- 53 Clark A P and Ghani F Detection processes for orthogonal groups of digital signals IEEE Trans Commun vol COM-21 1973 pp 907-915
- 54 Klein T J and Wolf J On the use of channel introduced redundancy for error correction IEEE Trans Commun vol COM-19 1971 pp 396-402
- 55 Mitten L G Composition principles for synthesis of optimal multistage processes Operations Research vol 12 1964 pp 610-619
- 56 Nemhauser G L Dynamic programming John Wiley and Sons Inc New York 1966
- 57 Hammer P L and Rudeanu S Boolean methods in operations research Springer Verlag New York 1968
- 58 Wang C L Best least square approximation by minimum property of Fourier expansions Theory of approximations with applications Ed Alan G L and Sahancu B N Academic Press 1976 pp 86-110
- 59 Hunt B R The inverse problem of radiography Math Biosciences vol 8 1970 pp 161-179
- 60 Hunt B R Deconvolution of linear systems by constrained regression and its relationship to Wiener theory IEEE Trans Auto Contr vol AC-17 1972 pp 703-705
- 61 Commenges D The deconvolution problem fast algorithms including the preconditioned conjugate-gradient to compute a MAP estimator IEEE Trans Auto Contr vol AC-29 1984 pp 229-243
- 62 Rosenfield A and Kac C Digital picture processing Second Edition vol 1 Academic Press New York 1982

- 63 Dines K A and Kac C Constrained least squares filtering IEEE Trans Acoust Speech and Process vol ASSP 25 1977 pp 346-350
- 64 Picchi G and Prati G Self orthogonalizing adaptive equalization in the discrete frequency domain IEEE Trans Commun vol COM-32 No 4 April 1984 pp 371-379
- 65 Pratt W K Fast computational techniques for pseudo-inverse and Wiener image restoration IEEE Trans Computers vol C 26 1977 pp 571-580
- 66 Narayan S S and Peterson A M Frequency domain least mean square algorithm IEEE Proc vol 69 1981 pp 124-126
- 67 Arya V K and Aggarwal J K Deconvolution of seismic data Hutchinson Ross Publishing Company Stroudsburg 1982
- 68 Jansson P A Deconvolution with applications in spectroscopy Academic Press Inc New York 1984
- 69 Phillips D L A technique for the numerical solution of certain integral equations of first kind J Ass Comput Mach vol 9 1962 pp 84-97
- 70 Twomey S The application of numerical filtering to the solution of integral equations of the first kind encountered in direct sensing measurements J Frank Inst vol 279 1965 pp 95-109
- 71 Rice R B Inverse convolution filters Geophysics vol 27 1961 pp 4-18
- 72 Davio M Deschamps J P and Andre T Discrete and switching functions McGraw Hill New York 1978
- 73 Wolf J K Efficient maximum likelihood decoding of linear block codes using a trellis IEEE Trans Inform Theory vol IT-24 1978 pp 76-80
- 74 Ralston A and Rabinowitz P A first course in numerical analysis, McGraw-Hill Tokyo 1978
- 75 Trench W F An algorithm for the inversion of finite Toeplitz matrices J Soc Indust Appl Math vol 12 1964 pp 515-522

- 76 Durbin J Efficient estimation of parameters in moving-average models Biometrika vol 46 1959 pp 306-316
- 77 Bennet W R Synthesis of active networks Proc of the symposium on modern network synthesis Polytechnique Institute of Brooklyn 1955 pp 45-61
- 78 Trench V F Inversion of Toeplitz band matrices Math Computation vol 28 1974 pp 1069-1095
- 79 Dickinson R J Efficient solution of linear equations with banded Toeplitz matrices IEEE Trans Acoust Speech Signal Process vol ASSP-27 1979 pp 421-423
- 80 Jain A K Fast inversion of banded Toeplitz matrices by circular decompositions IEEE Trans Acoust Speech Signal Process vol ASSP 26 1978 pp 121-126
- 81 Clin R E Plemmons P J and Jorj G Generalized inverse of certain Toeplitz matrices LA and its applications vol 8 1974 pp 25-33
- 82 Davis P J Circulant matrices John Wiley and Sons New York 1979
- 83 Cooley J W Lewis P and Welch P Application of the fast Fourier transform to the computation of Fourier integrals Fourier series and convolutional integrals IEEE Trans Audio Electroacoustics vol AU-15 1967 pp 79-84
- 84 Cooley J W and Tukey J W An algorithm for the machine calculation of complex Fourier series Math Computation vol 19 1965 pp 297-301
- 85 Penrose R A generalized inverse for matrices Proc Cambridge Phil Soc vol 51 1955 pp 406-413
- 86 Penrose R On best approximate solutions of linear matrix equations Proc Cambridge Phil Soc vol 52 1956 pp 17-19
- 87 Campbell S L and Meyer C D Jr Generalized inverses of linear transformations Pitman Publishers London 1979
- 88 Godard D Channel equalization using a Kalman filter for fast data transmission IBM J Res Dev vol 18 1974 pp 267-273

- 89 McCormick G P Nonlinear programming theory
algorithms and applications John Wiley and Sons
New York 1983
- 90 Goldstein A A Constructive real analysis Harper and
Row New York 1967
- 91 Rosen J B The gradient projection method for nonlinear
programming Part I SIAM J Appl Math 1960 pp 181-218
- 92 Kobayashi H Application of Hestens-Steifel algorithm
to channel equalization Conference record IEEE Inter
national Conference on Communications Montreal 1971
- 93 Kuhn H W and Tucker A W Linear inequalities and
related systems Annals of Mathematical Studies No 38
Princeton Univ Press 1956
- 94 Kretzmer E R Generalization of a technique for binary
data communication IEEE Trans Commun vol COM-14 1966
pp 67-68
- 95 Burkhardt H and Barbosa L C Contributions to the
application of the Viterbi algorithm IEEE Trans Inform
Theory vol IT-31 1985 pp 626-634

95-65

E-1986-D-P00 FIN